

Benutzerhandbuch Automatisierungs- und Fernwirksystem @120

Version 3.7

Lizenzbestimmungen

Die in diesem Handbuch beschriebene Software wird Ihnen gemäß den Bedingungen eines Lizenzabkommens zur Verfügung gestellt und darf nur unter den darin beschriebenen Bedingungen eingesetzt werden.

Daten, Abbildungen, Änderungen

Daten und Abbildungen sind unverbindlich. Änderungen, die dem technischen Fortschritt dienen, sind vorbehalten. Falls Sie Verbesserungs– oder Änderungsvorschläge haben oder Fehler in dieser Druckschrift entdecken sollten, bitten wir um Ihre Mitteilung.

Gewährleistung

KEINE GEWÄHRLEISTUNG. Die technische Dokumentation wird Ihnen OHNE Mängelgewähr geliefert. OHP übernimmt keine Gewährleistung für deren Genauigkeit oder Verwendung. Die Verwendung der technischen Dokumentation oder der darin enthaltenen Informationen hat der Benutzer zu verantworten.

Schulung

Zur Vermittlung ergänzender Systemkenntnisse werden von OHP entsprechende Schulungen angeboten.

Warenzeichen

OHP, das OHP-Logo und ProWin sind Marken der OHP GmbH.

Die übrigen in diesem Handbuch verwendeten Produktnamen können eingetragene Warenzeichen und/oder Warenzeichen der jeweiligen Unternehmen sein.

Copyright

Gedruckt in Deutschland.



Inhalt





Teil I Allgemeines

1 Gültigkeitshinweis

Diese Beschreibung gilt für die ALU 320x ab Firmwareversion 3.68 und MP@Plus Version 2.8. Die aktuelle Firmwareversion stellen wir auf www.ohp.de im Kundenbereich als Download zur Verfügung.

Teilleistungen werden möglicherweise von älteren Firmwareversionen nicht unterstützt. Eine Aufstellung der Firmwareversionen und die zugehörigen neuen Leitungen finden Sie in **TEIL II Projektierung und Programmierung,** Kapitel 1.

2 Einführung

Das Automatisierungs- und Fernwirksystem @120 versteht sich als Weiterentwicklung der bewährten A120/Compact Automatisierungs- und Fernwirktechnik und sichert die langfristige Verfügbarkeit der A120/Compact Baureihe als leistungsfähige Automatisierungsplattform.

Das Automatisierungs- und Fernwirksystem @120 kann neben der Ausrüstung von Neuanlagen insbesondere auch zur Migration und damit zur Investitionssicherung von vorhandenen Anlagen eingesetzt werden, da sich das System nahtlos in die E/A Welt der A120 /Compact Baureihe einfügt.

2.1 Systemneutrale Automatisierungsplattform

Das @120 System ist als systemneutrale Automatisierungsplattform ausgelegt, die auf herstellerspezifische Lösungen soweit als möglich verzichtet. Basierend auf modernster offener embedded Industrie-PC Technik mit dem Standard Echtzeit Betriebssystem Kernel Windows CE werden alle PC Standard-Schnittstellen wie z.B. Ethernet, RS232, USB, VGA, CF-Card unterstützt.

Mit der herstellerunabhängigen IEC 61131-3 Programmiersoftware MULTIPROG ist ein leistungsfähiges und ausgereiftes Programmiertool implementiert, welches bereits heute von über 70 namhaften SPS Firmen erfolgreich eingesetzt wird.

Mittels genormter Kommunikationsprotokolle wie der IEC 60870-5-101/104, oder dem weit verbreiteten ModbusTCP Protokoll und auch den optionalen OPC Server kann das System in verschiedenste Automatisierungs- bzw. Fernwirkumgebungen integriert werden.

Über die integrierte Ethernet- bzw. CAN-Schnittstelle können systemeigene E/A Erweiterungen wie die DEA 300 oder auch handelsübliche E/A Systeme z.B. von Phoenix Contact angeschlossen werden. Über diese E/A-Erweiterungen können so maximal bis zu 1.216 E/A Punkte verarbeitet werden.

2.2 Migration von AEG/Schneider Automatisierungs- und Fernwirktechnik

Die A120/Compact Geräte bilden eine bewährte Automatisierungs- und Fernwirkplattform, in deren Installation erhebliche Investitions- und Zeitaufwendungen geflossen sind. Mit dem @120 System hat OHP eine Strategie entwickelt, die bei minimierten Kosten und Zeitaufwendungen die A/U-Geräte in eine neue und langfristig verfügbare Automatisierungs- und Fernwirkbaureihe migriert.

Bestechender Vorteil dieser Strategie ist die Kostenersparnis durch Nutzung vorhandener Peripherie wie E/A-Module und KOS-Baugruppen mit vorhandener Verdrahtung: Es wird nur ausgetauscht, was ausgetauscht werden muss. Dies sind in der Regel nur die ALU und der Hauptbaugruppenträger (DTA200 -> DTA300).

Durch diese Maßnamen stehen die neuen genormten IEC 60870-5-101/-104 Prozeduren mit moderner Programmiertechnik unter IEC 61131-3 zur Verfügung. Durch die Nutzung der vorhandenen KOS-Baugruppen ergebenen sich elegante Upgradestrategien von 1/F- und 1/W-Prozeduren auf genormte IEC-Prozeduren.

3 Systemvoraussetzungen

3.1 Systemvoraussetzungen Programmiersoftware

Hardware

Für die optimale Leistungsfähigkeit des SPS-Programmiersystems MULTIPROG müssen mindestens die folgenden Hardware-Voraussetzungen erfüllt sein:

Gerät/Baugruppe	Minimal	Empfohlen ab
IBM-kompatibler PC mit Pentium-Prozessor	Pentium II 350 MHz	Pentium III 500 MHz
Arbeitsspeicher	64 MB	128 MB
Festplatte	250 MB freier Speicherplatz	
CD-ROM-Laufwerk	benötigt	
VGA-Monitor Farbeinstellungen Auflösung	256 Farben 800 x 600	True Color 1024 x 768
Ethernet-Schnittstelle	benötigt	

Software

MP@Plus benötigt folgendes Betriebssystem:

- Microsoft Windows 2000
- Microsoft Windows XP



3.2 Systemvoraussetzungen Steuerung

- Als Hauptbaugruppenträger muss das DTA 300 verwendet werden.
  Bei Umrüstung bestehender A120/Compact Stationen muss das DTA 200 ausgetauscht werden.
- Steckplatz 1 ist ausschließlich für ALU 320 reserviert
- Als Erweiterungsbaugruppenträger kann ein DTA 301, DTA 201 oder DTA 202 verwendet werden.
- Es werden die Standard E/A-Karten der A120/Compact unterstützt (siehe Tabelle).
- Die OHP E/A-Baugruppen der 300er Serie können eingesetzt werden (siehe Tabelle).
- Über ModbusTCP Client können externe E/A-Komponenten und Systemerweiterungen z.B. von Phoenix Contact, ABB angeschlossen werden.

3.2.1 @120 Systemkomponenten und E/A Vergleichsliste

Nachfolgend sind die Systemkomponenten für das Automatisierungs- und Fernwirksystem @120 aufgeführt.

Insbesondere auch die E/A Baugruppen, die klemmenkompatibel zu den Schneider E/A-Baugruppen sind.

Die aufgeführten OHP E/A-Baugruppen sind z. Zt. noch nicht vollständig verfügbar. OHP behält sich bei den E/A-Baugruppen vor, Vergleichstypen von Schneider Electric zu liefern. E/A-Baugruppen, die sich nicht in der Liste befinden, können ggf. auf Anfrage geliefert bzw. integriert werden.

Bezeichnung	**Schneider Vergleichstyp**	**Beschreibung**	
		Zentraleinheit, Grundbaugruppenträger	
ALU320		Zentraleinheit für @120 System	
DTA300		Grundbaugruppenträger für Zentraleinheit und 4 E/A Baugruppen	
DTA301	DTA201	Erweiterungsbaugruppenträger für 5 E/A Baugruppen	
MP@Plus		MULTIPROG IEC 61131-3 Programmiersoftware für @120 System. AWL, FBS, KOP,ST und AS. Querübersetzung AWL-KOP-FBS, Offline Simulation, Online Change, Watch Fenster, Debugfunktionen, Projektvergleich, Querverweise.	
		Digitale Eingabe	**Eingangsspannung**
DEP314	DEP214	16 Digitale Eingänge	12V bis 60V DC
DEP316	DEP216	16 Digitale Eingänge	24V DC
DEP332		32 Digitale Eingänge	24V DC
		Digitale Ausgabe	**Ausgangsspannung**
DAP316	DAP216N	16 Digitale Ausgänge	24V DC

DAP332		32 Digitale Ausgänge (Hardware in Vorbereitung)	24V DC
DAP308	DAP208	8 Relaisausgänge	24…110V DC oder 24…230V AC
		Digital kombinierte Baugruppen	**Eingangsspannung** **Ausgangsspannung**
DAP312	DAP212	8 Digitale Eingänge 4 Relais Ausgänge	24V DC 24…110V DC oder 24…230V AC
DAP320	DAP220	8 Digitale Eingänge 8 Digitale Ausgänge	24V DC 20…30V DC
		Analoge Eingabe	**Eingangssignal**
ADU304	ADU204	4 analoge Eingänge, 12Bit + VZ	+/- 500mV, PT100
ADU306	ADU206	4 analoge Eingänge, potentialgetrennt, 11Bit + VZ	+/- 10V, +/- 20mA
ADU308		8 analoge Eingänge, 12Bit + VZ	+/- 10V, +/- 20mA
ADU314	ADU 214	8 analoge Eingänge, 15Bit + VZ	+/- 10V, +/- 20mA Temperatur- und Widerstandsmessung
		Analoge Ausgabe	**Ausgangssignal**
DAU302	DAU202	2 analoge Ausgänge, potentialgetrennt, 11Bit + VZ	+/- 10V, +/- 20mA
DAU304	DAU204	4 analoge Ausgänge, potentialgetrennt, 11Bit + VZ	+/- 1,5V, 10V, 20mA, 4…20mA
DAU308	DAU208	8 analoge Ausgänge, potentialgetrennt, 11Bit + VZ	+/- 10V
		Koppelbaugruppen	**Schnittstelle**
KOS202A		Kommunikationsmodul für Fernwirkprotokoll Modnet 1/F.	1 x RS232
KOS203		Kommunikationsmodul für Fernwirkprotokoll Modnet 1/W.	1 x RS232
KOS260		Kommunikationsmodul für Fernwirkprotokoll IEC 60870-5-101	2 x RS232



4 Kommunikationsschnittstellen des @120 Systems

Das neue Automatisierungs- und Fernwirksystem @120 beinhaltet Kommunikationsschnittstellen, die entweder auf der ALU 320 direkt oder über KOS-Koppelbaugruppen betrieben werden können.

4.1 Kommunikationsprozeduren auf der ALU320:

- IEC 60870-5-101
- IEC 60870-5-101 AWD für **A**utomatischen **W**ähl**d**ienst (in Vorbereitung)
- IEC 60870-5-104
- ModbusTCP Client
- Modbus RS232 Slave

Die IEC Prozeduren entsprechen der Kompatibilitätsliste @120 für IEC 60870-5-101 und IEC 60870-5-104.

4.2 Kommunikationsprozeduren mit KOS Koppelbaugruppen

- Modnet 1/F (SEAB-1F) - KOS 202A
- Modnet 1/W - KOS 203
- Modnet 1/W AWD - KOS 203
- IEC 60870-5-101 - KOS 260
- IEC 60870-5-101 AWD - KOS 260



Hinweis: Die KOS 201 bestehender Stationen wird unterstützt.

4.3 Modembaugruppen für Fernwirkprozeduren

Für das @120 System sind über die Kommunikationsprozeduren IEC 60870-5-101 – AWD und IEC 60870-5-104 folgende Modemankopplungen möglich:

- UEM001 - FSK Standleitungsmodem, 600/1200 Baud, CCITT-Raster.
  Einbauplatine für KOS 202, KOS 203 oder KOS 260
- UEM201 - FSK Standleitungsmodem, wie UEM 001, jedoch Becherform für Montage auf DTA300/301
- UEM202 - FSK Standleitungsmodem, wie UEM 001, jedoch Hutschienenmontage
- UEM300 / UEM301 - Analog-Modem für Wählleitungsbetrieb
- UEM302 - ISDN-Terminal für Wählleitungsbetrieb
- UEM303 / UEM 304 - Dualband GSM Modem
- UEM306 / UEM 307 - GPRS Modem

4.4 Ankopplung über Modbus

4.4.1 ModbusTCP Client

Die ALU 320 ist mit einer ModbusTCP Client-Schnittstelle ausgerüstet, über die externe E/A Module oder Steuerungen die über eine ModbusTCP Server-Schnittstelle verfügen, angekoppelt werden können.

Anmerkung: Der Client entspricht in seiner Funktionalität dem Master bei einer seriellen Modbus-Ankopplung.

Weiterführende Informationen siehe **Teil II Projektierung und Programmierung,** Kapitel 4.4.

4.4.2 Modbus RS232 Slave / Modbus TCP-Server

Die zwei seriellen Schnittstellen der ALU 320 können als Modus Slave Schnittstelle projektiert werden. Der Modbus TCP-Server wird über die Ethernet-Schnittstelle angekoppelt.

Über diese Schnittstellen können beliebige Modbus-Master / Modbus-Clients Daten aus der Station abfragen. Bis auf die Auswahl des Interfaces (SERIELL oder TCP_IP) ist die Projektierung für beide Protokolle identisch.

Von der ALU 320 werden folgende Funktionscodes unterstützt:

FC 1	Lese Bit	Read Coils
FC 2	Lese Eingang	Read Discrete Inputs
FC 3	Lese Register	Read Holding Register
FC 4	Lese Eingangswort	Read Input Register
FC 5	Schreibe einzelnes Bit	Write Single Coil
FC 6	Schreibe einzelnes Register	Write Single Register
FC 15	Schreibe Bitbereich	Write Multiple Coils
FC 16	Schreibe Registerbereich	Write Multiple Register
FC 23	Lese/Schreibe Registerbereich	Read/Write Multiple Register

Die Daten werden im Shared Memory Bereich zur Verfügung gestellt. Die Projektierung erfolgt über INI-Dateien, die auf die CF-Karte gespeichert werden.

Anmerkung: Die Funktionen FC 2 und FC 4 greifen nicht direkt auf die Eingänge der Hardware zu. Die Eingänge müssen auch hier auf dem Shared Memory Bereich abgebildet werden.

Anmerkung: Der Modbus TCP-Server unterstützt maximal 8 Client-Verbindungen. Hierbei ist zu beachten, dass der Modbus-Client einer @120 oder @250 jeweils eine Verbindung für das Lesen und eine für das Schreiben von Daten herstellt. Die Anzahl der Transaktionen (gleichzeitig gesendete Abfragen bei asynchronem Betrieb) ist auf 4 begrenzt.



Weiterführende Informationen siehe **Teil II Projektierung und Programmierung** Kapitel 4.5, Kapitel 6.3 und Kapitel 6.5.



TEIL II Projektierung und Programmierung

Die Projektierung und Programmierung wird Anhand eines Projekts erläutert, das mit der Vorlage "ALU320_KOS202" erstellt wurde.

Diese Vorlage kann von der Internetseite der Firma OHP Automation Systems GmbH geladen werden.

Die Vorlage beinhaltet folgend Beispiele:

- die I/O Konfiguration für eine Eingabebaugruppe DEP 316 und eine Ausgabebaugruppe DAP 316
- Datentyp-Deklaration eines Arrays von 128 Byte, das für die KOS 201/202 verwendet werden kann
- einen Funktionsbaustein für die Bildung von umlaufenden 16 Bit Zählwerten für digitale Eingänge
- ein Programm "Main" als Funktionsplan, in dem die Baugruppen-Ausfallmerker für die DEP 316 und DAP 316 zu einem Sammelfehler zusammengefasst werden.
- die globalen Variablen für Statusinformationen der PAB-Teilnehmer



1 Neue Leistungen der ALU 320 Firmware

Die aktuelle Firmware-Version kann über das Multiprog Menü "Online – Projektkontrolle – Info" ermittelt werden.



Hinweis: Da die Treiber für ALU 320 und ALU 5xx zum Teil aus gleichen Quellprogrammen gebunden werden, kann sich der Versionsindex erhöhen, auch wenn die jeweiligen ALU-Treiber von dieser Änderung nicht betroffen sind. Aus diesem Grund kann es zu Versionssprüngen kommen.

Version 1.40

Es wurde eine zusätzliche Funktionalität bei den Treibern IEC 60870-5-101 und ...104 eingeführt. Für Sollwerte und Bitstrings in Befehlsrichtung kann jetzt ein Initialwert vorgegeben werden. Dieser Initiallwert kann auf zwei Arten verarbeitet werden:

Modus 1:
Der projektierte Initialwert wird einmalig bei Kaltstart in die entsprechenden %IW bzw. %ID Variablen eingetragen.
Sobald ein Telegramm vom Leitsystem eintrifft, wird der Wert des Telegramms solange in die entsprechende Variable eingetragen, bis ein neuer Wert vom Leitsystem eintrifft.

Modus 2:
Der projektierte Initialwert wird bei Kaltstart in die entsprechenden %IW bzw. %ID Variablen eingetragen.
Trifft ein Telegramm vom Leitsystem ein, wird der Wert des Telegramms in die entsprechende Variable eingetragen und anschließend wieder durch den projektierten Defaultwert überschrieben. Je nach Länge des Anwenderprogramms kann der Telegrammwert für 1 ... 5 Programmzyklen anstehen, bevor er durch den Initialwert überschrieben wird.
Auf diese Weise kann vom Anwenderprogramm erkannt werden, ob ein neues Telegramm vom Leitsystem geschickt wurde, auch wenn sich der Wert im Telegramm selbst nicht geändert hat.

Zwischen Modus 1 und 2 kann ebenfalls projektierbar gewählt werden.

Projektierungsbeispiel:
```
//: Bitstring C_BO_NA
1:VAR_CBO_NA_INIT= 0x0000FFFF
1:VAR_CBO_NA_CMD= TRUE     // TRUE = Modus 2, Initialwert nach jeder Telegrammübergabe

//: Sollwert C_SE_NA
1:VAR_CSE_NA_INIT= 0xFFFF
1:VAR_CSE_NA_CMD= FALSE     // FALSE = Modus 1, Initialwert nur bei Kaltstart
```

Version 1.54

Uhrzeitsynchronisation über GPS, IEC...101/104 Zeittelegramme und Anwenderprogramm.

Neue Parameter für SVC_RTUn.INI

RTU_TIME_SET: 0, 1 oder 2

0 = Zeit darf über IEC gestellt werden (default)
1 = Zeit darf nie über IEC gestellt werden
2 = Zeit darf nur über IEC gestellt werden, wenn aktuelle Zeit IV (ungültig en: invalid) ist

Der folgende Parameter ist nur gültig, wenn RTU_TIME_SET ungleich NULL ist!

RTU_TIME_ACK_NEG: FALSE oder TRUE



FALSE = IEC Zeittelegramm wird mit positiver Response bestätigt (default)
TRUE = IEC Zeittelegramm wird mit negativer Response bestätigt

Der folgende Parameter ist nur gültig, wenn RTU_TIME_SET = 2 ist!

RTU_TIME_SYNC_ENABLE: Eingabe in Minuten (1…65535)
Delta-Zeitangabe in Minuten, seit dem letzten Stellen der Uhr. Nach Ablauf darf die Uhr über IEC-Telegramm gestellt werden.

Version 1.61
Wenn ein ADU projektiert aber nicht gesteckt ist, wird der Messwerte jetzt mit 0 angezeigt (früher 32767).

Version 1.62
Treiber für ADU204/205 Karten geändert

Version 1.70

Die Namen der INI-Dateien wurden geändert, damit diese mit der neuen Bedienoberfläche über Multiprog erzeugt werden können.

Alter Name	Neuer Name ab Version 1.70
PCOS_A320.ini	PCOSA320.ini
Slot_Device.ini	SlotDev.ini
SVC_COMn.ini *)	SVCCOMn.ini *)
SVC_RTUn.ini *)	SVCRTUn.ini *)
Modbus_Server.ini	MBSRV.ini

*) n steht für die jeweils zugehörige Service-Routine (1…4)

Achtung: Wenn Sie eine neue Firmware ab Version 1.70 verwenden und Ihre INI-Dateien von älteren Firmwareversionen benutzen möchten, müssen Sie diese vor dem Speichern auf der CF-Karte umbenennen. Inhaltlich sind die Dateien gleich geblieben.

Version 1.74

Es können jetzt beide COM-Schnittstellen gleichzeitig für die IEC 60870-5-101 Kommunikation verwendet werden.

Es können jetzt zwei Service-Routinen für die Kommunikation mit IEC 60870-5-104 definiert werden.

Weiterführende Informationen siehe **Teil II Projektierung und Programmierung,** Kapitel 6.4.2.

Version 1.76
Es können jetzt mehrere KOS 304 verwendet werden (Steckplatz 2 bis 5).

Version 1.80
Firmwarebaustein für RS232 (SCOM) Kopplung eingebracht.
Bei Blocktelegrammen IEC...101/104 werden jetzt nur noch max. 127 Objekte vom Typ Einzel- oder Doppelmeldung geblockt.

Grenzen für Übergabeschnittstelle erweitert

IEC101
Maximal 2096 Objekte in Melde- und Befehlsrichtung in der SVCRTUn.INI Datei.
Maximal 512 Byte in Befehlsrichtung und 1024 Byte in Melderichtung für die I/O-Konfiguration

IEC104
Maximal 6000 Objekte für Service 1 und 2 und max. 2096 Objekte für alle weiteren Service-Routinen in der SVCRTUn.INI Datei.
Maximal 512 Byte in Befehlsrichtung und 1024 Byte in Melderichtung für die I/O-Konfiguration

Version 1.81 - 1.86
Einbinden der Treiber für das integriete GPRS-Modem und diverse Modifikationen.
Projektierung der Schnittstelle über Datei ALURAS.ini

Version 1.81: Sommer/Winterzeit Verwaltung per Uhrzeittelegramme IEC104/101 überarbeitet

Version 1.87

- IEC101 AWD Betrieb: Routine für das Senden des Passworts geändert. Die Zeichen werden jetzt schneller nacheinander gesendet.
- Modbus TCP für mehrere Request freigegeben
- System Shared Memory auf max. 32KByte erweitert!
- Modbus Shared Memory auf max. 32KByte erweitert!
- PAB520.dLL - BIK114 Treiber für SEAB erweitert
- SSDKOSBIKA.dll und CDLSSDKOSBIKA.dll für SEAB-FWB erweitert
- SEAB Firmwarebibliothek für Multiprog bzw. ProConOS eingebunden

Version 1.89

- SSDKOSBIKA.dll und CDLSSDKOSBIKA.dll für SEAB-FWB erweitert
- IEC101/104: Ringpufferdaten werden bei Unterspannung auf CF-Karte geschrieben und bei Warmstart wieder geladen
- IEC101/104: Behandlung des OV-Bit (Overrange) bei Messwerten überarbeitet
- IEC101: Objekt-Nr. jetzt auf 3 Oktett erweitert

Version 1.90

- IEC101/104 ADU Messwertverarbeitung auf 2er Komplement umgestellt und Qualifier OV Anstoß Übertragung geändert
- IEC101 AWD Betrieb:
  - AT_INIT String auf 28 Zeichen erweitert
  - 2 weitere AT_INIT Strings für SVCCOMn.ini zugelassen (AT_INIT2 und AT_INIT3)
- SEAB-FWB - Start/STOP Problem nach AWP Download beseitigt



Version 1.91

- A520/A320 Modbus Server TCP/IP Kommunikation für mehrere Transaktionen optimiert.
- IEC101 AWD LED Ansteuerung eingebunden:
  Die Funktion ist nur für ein internes MODEM (COM3) vorgesehen.
  Einstellung in SVCCOM?.INI unter Section AWD101-MODE
  MODEM_LED=TRUE/FALSE

  Auswirkung bei TRUE:
  LED ON - Datenverbindung ("CARRIER") vorhanden - Passwort (Rx/Tx) und IEC-Kommunikation.
  LED OFF - keine Verbindung
  LED Blinken(Slow) - Verbindungsaufbau UST->LR
  LED Blinken(Fast) - Verbindungsaufbau LR (Ring)->UST
  LED 2 mal blinken (500ms) und Pause ... Modem Initialisierungsproblem
  LED 3 mal blinken (300ms) und Pause ... Modem nicht mehr bereit
- Bei IEC101-AWD das Senden des Passworts mit "CR" abschließen (default: TRUE)
  Deaktivieren dieser Funktionalität unter Section AWD101-MODE mit DIAL_PW_CR=FALSE

Version 1.94

- VPN - GPRS Modem HW-RESET für internes Modem implementiert.
  Diese Funktionalität ist jetzt als Standardeinstellung implementiert und kann in ALURAS.ini Section AX-RASVPN-PARAMS wie folgt abgeschaltet werden: HwReset3Att=FALSE
- Bei Telnet Verbindung - Willkommenstext mit Versionskennung implementiert

Version 3.00 (BETA Projektversion)

- Umstellung auf PROCONOS4 auf Basis der Version 1.94.
  (Voraussetzung für Online-Ändern)
- BIK Geräte Treiber für A520/A320 modifiziert

Version 3.10 (BETA Projektversion)

- Codebasis für SEAB und DEA Betriebsart von A520/A320 angepasst
- AWD Betrieb:
  Ablauf InitString mit Motorola Modem korrigiert
  Wartezeit für Wahlwiederholung korrigiert

Version 3.12 (BETA Projektversion)

- Auswertung der ALUX20.ini für SNTP Parameter "Server2" ergänzt.
  Wichtig: Der Parameter "SERVER2" muss immer hinter "SERVER" angeeben werden!
- Datum/Uhrzeit Überwachung für SNTP Dienst erweitert
- Diag_TSync.exe
  Diagnose für Datum/Uhrzeit Dienst SNTP Client (NTP0) etc. SNTP - 2 Serveradresse freigegeben.
- SNTP Service (Client).
  Zweite Serveradresse ergänzt. Die Serveradresse auf IP Adressebasis (xxx.xxx.xxx.xxx) erweitert, zuvor war nur Hostname möglich.
- UDPCFG Server
  Aktualisierung der TCP/IP Adresse. Uhrzeit stellen mittels Broadcast erlaubt!

Version 3.16

- BIK DEA Ankopplung wegen BKF (Interbus-S) Modul modifiziert.
- Offizielle Freigabeversion der neuen Leistungen ab 3.0

Version 3.19

- Modbus TCP-Client
- Priorität der Modbus Threads auf PCOS Systemtask Niveau (121) angepasst.
- Verbindungsaufbau zu Modbus Servern optimiert (REACT).
- Überwachung Projektierungsparameter eingeführt.
- Überwachung der Response-Timeout Zeiten mit 10ms Faktor beaufschlagt.
- Anzahl max. Modbus Server auf 32 IP Adressen begrenzt.
- Anzahl I/O Gruppen auf max. 100 überwacht.
- Ablauf im Successive Mode (Response TOUT) überarbeitet.

Version 3.20

- Systemmerker Baugruppenstörung und Slot-Status (DWORD) für BIKX14 Modnet 1/SFB Status integriert.
- BIKx14 Ausfallmerker für reine Modnet 1/SFB Kommunikation überarbeitet.

Version 3.21

- Modbus TCP/IP Server: Anzahl Verbindungen von 9 auf16 erhöht.

Hinweis: KW-Client benötigt 2 Verbindungen, d.h. eine für Read-Requests und eine für Write-Requests.

Version 3.40

- SystemTick geändert

  Parameter "KSYSTICKDIS= (TRUE/FALSE)" für PCOSAX20.ini in Sektion PCOS-SYSTEM aktiviert (def. FALSE) .
  KSYSTICKDIS = TRUE bewirkt, dass der Systemtick sich wie in den vorhergehenden Versionen verhält, d.h. zyklische Tasks brauchen die doppelte Intervallzeit!
  KSYSTICKDIS = FALSE bewirkt, dass der Systemtick halbiert wird, dies bewirkt, dass zyklische Tasks genau mit der projektierten Intervallzeit aufgerufen werden.
  Ist der Parameter aktiv, so erscheint in der Multiprog Versionsanzeige hinter V3.40 ein "S" (V3.40S) für ProConOS ohne Kernel SystemTick.

Version 3.41

- ALU 310 eingeführt

  ALU310 = 32 kB Programmspeicher, 32 kB Data, 4 kB Retain, 2 kB Shared Memory
  ALU320 = 1 MB Programmspeicher, 1MB Data, 4 kB Retain, 32 kB Shared Memory



Version 3.43

- Firmwarebibliothek ALUX20_SBP_SEAB1N für SEAB1N Kommunikation eingeführt (zur Zeit nur Slave Funktionalität).

Version 3.44

- Systemmerker Baugruppenstörung und Slot-Status (DWORD) für BIKX14 Modnet 1/SFB Status integriert.
- BIKx14 Ausfallmerker für reine Modnet 1/SFB Kommunikation überarbeitet.

Version 3.45

- IEC 60870-5-103 eingeführt
  Modul mit IEC101 Datentypen im privaten Bereich der IEC103 realisiert.

Version 3.50

- Neue ProConOS Version (4.0.494) von KW-Software eingebunden. Verbesserung beim Online-Tauschen.

Version 3.60

- Hardware-Watchdog über Anwenderprogramm
- Erweiterung IP-Alias. Mit dieser Funktion kann pro Netzwerkinterface eine zweite IP Adresse zugeordnet werden. Der Eintrag erfolgt in ALU320.INI

Version 3.62

- PAB320 Treiber für DAP332 erweitert.
- Hochlaufverfahren geändert, deshalb ein etwas verlängerter Boot-Phase. Bei Änderungen von HOMEDST/AUTODST in ALUX20.ini wird ein 2. Boot-Vorgang eingeleitet.

Version 3.64

- Modbus TCP/IP: Durch Netzwerkstörungen (WLAN/VPN) verursachte offene Verbindungen (inaktiver Port) werden jetzt geschlossen.

Version 3.65

- Modnet 1N Slave - Bus Synchronisation korrigiert.
- Neue Firmware-Bibliothek ALUX20_OHP4_SYS_OS eingebunden. Die Bibliothek stellt mehrere Bausteine für die Steuerung des RASVPN Dienst (AXR1:) und ein Reboot Kommando für die ALU der Anwendertask zur Verfügung.

  AX20_SOS_REBOOT
  Neustart(Warmstart) der ALU

  AX20_SOS_RAS_MODEMRESET
  HW-Reset (Kaltstart) für internes MODEM des RASVPN Dienst ausführen

AX20_SOS_RAS_DISC
Verbindungsabbau RASVPN Dienst ausführen

AX20_SOS_RASVPN_STATUS
Verbindungsstatus und Betriebszustand des RASVPN Dienst abfragen

- IEC101 AWD Betrieb: CR/LF Strings angepasst
  Sektion [AWD101-MODE] BOOL Parameter "LINEFEED" eingefügt, LINEFEED abschalten (def. FALSE)
- Die Applikation DIAG_RASVPN.exe (V1.3) wurde um die Funktion "/TYP" erweitert, dies ermöglicht die Ausgabe des MODEM Typs und der FW Version, wenn der AXRVPN Dienst aktiv ist.

Version 3.66

- Firmwarebibliothek für SEAB-1F-Slave integriert.

Version 3.67

- PAB Treiber für MSK351 I/O - Ausgabe bei Spannungswiederkehr korrigiert.
- Multiprog Firmware-Bibliothek ALU352_MODTCP_CLTSYS für Modbus/TCP Client eingebunden.
  FWB Bausteine dienen zur Kommunikationsdiagnose!

Version 3.68

- Firmware-Bibliothek ALUX20_OHP4_SCOM modifiziert, um Probleme beim seriellen Modbus-Master zu beheben.
- Prüfung PCOSAX20.ini Sektion [MODBUS-SERVER] Wert für SharedMemoryOffset von 0 bis 65535 erweitert.
- Prüfung Datei MBSRV.ini Sektion [IFn-SERV-MODBUS] für Wert STATUS_OFFSET korrigiert. Speicherprüfung Sharedmemory-Size plus Statuslänge von >= auf > geändert.
  Die MBRV.ini wurde immer als fehlerhaft interpretiert, wenn der Offset für den SPS-Status so eingestellt war, dass der Status das letzte Byte im reservierten Bereich genutzt hat (Eintrag in ERR-Datei = Fkt. CFG_ChkIFace(), Fehler (-4)).



2 Erzeugen eines neuen Projekts mit MULTIPROG

Beim Einrichten eines neuen Projekts gehen Sie folgendermaßen vor:

Schritt 1 Im Menü Datei "Neues Projekt…" anwählen.

Schritt 2 Vorlage "ALU320_KOS202" mit Doppelklick auswählen.

Schritt 3 Gegebenenfalls die IP-Adresse für die Verbindung zur ALU 320 ändern (Teil II, Kapitel 3).

Schritt 4 Projekt über Menü Datei "Projekt speichern unter…" mit Projektnamen abspeichern.

Schritt 5 Das neue Projekt kompilieren (F9 oder Menü "Code -> Make"). Dadurch werden die Datentypen und Funktionsbausteine in den jeweiligen Auswahlfenstern zur Verfügung gestellt.

Schritt 6 Definition der Ein- und Ausgaben vornehmen (Teil II, Kapitel 4).

Schritt 7 Logischer Programm-Organisationseinheiten (POE) erstellen.

Schritt 8 Die erstellten POEs in die Liste der auszuführenden Tasks eintragen.

Schritt 9 Programmcode erzeugen (F9 oder Menü "Code -> Make").

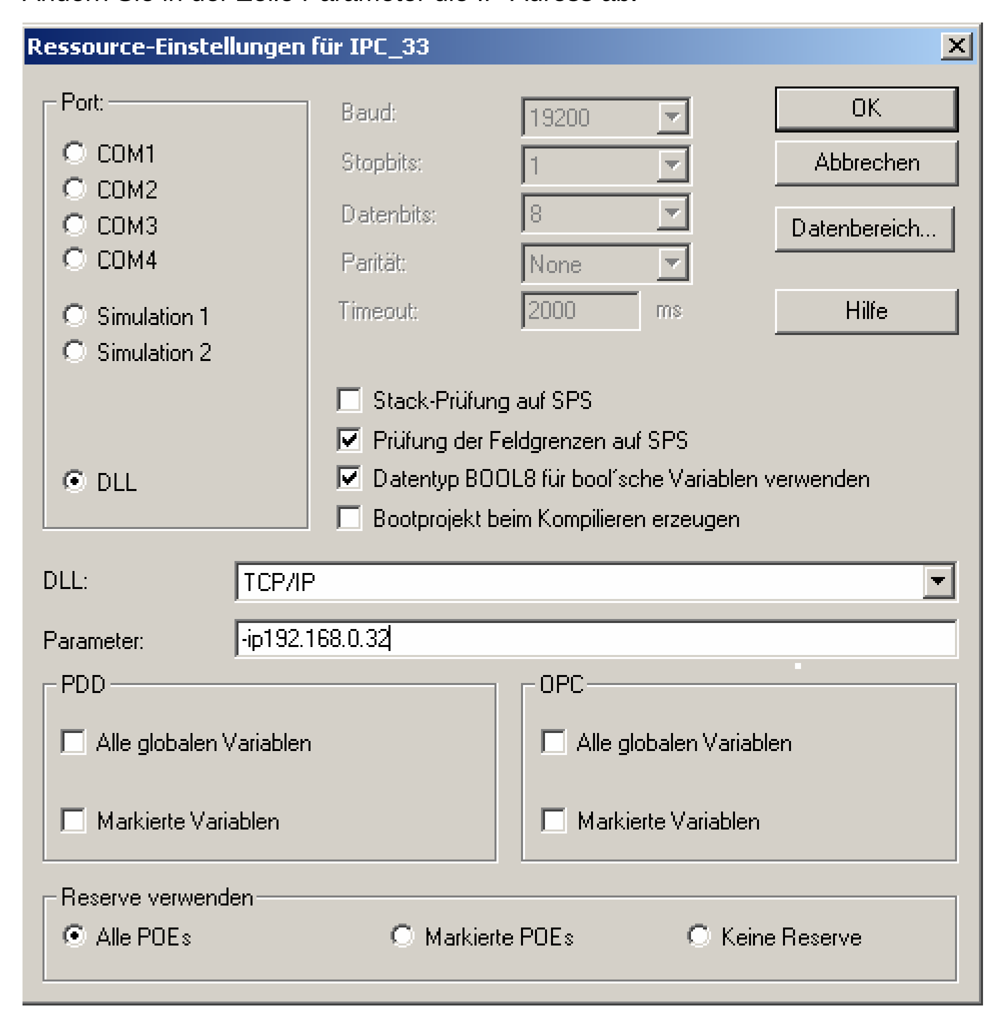
Schritt 10 Programmcode über das Menü "Online -> Projektkontrolle" in die ALU 320 senden und starten.

3 Ändern der IP-Adresse für die Verbindung zur ALU 320



Wählen Sie in der Hardwarekonfiguration die Ressource, im Beispiel ALU320:PCOS_CE, mit der rechten Maustaste an. Öffnen Sie über das Pop-UP-Fenster den Dialog "Einstellungen… ".

Ändern Sie in der Zeile Parameter die IP-Adress ab.





4 Definition der Ein- und Ausgaben

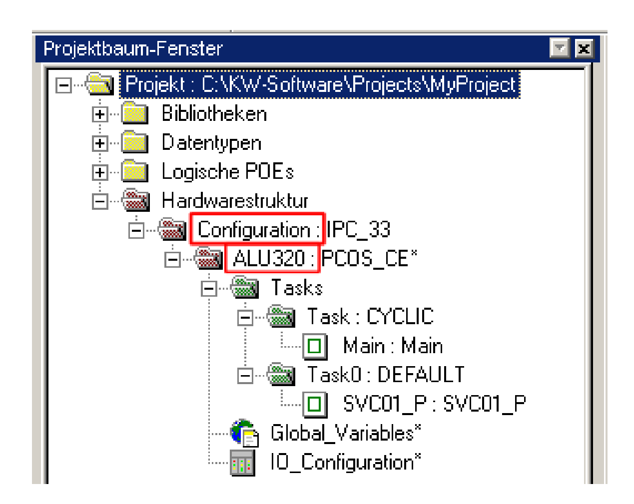
4.1 Ein- und Ausgaben mit dem OHP SPS-Konfigurator

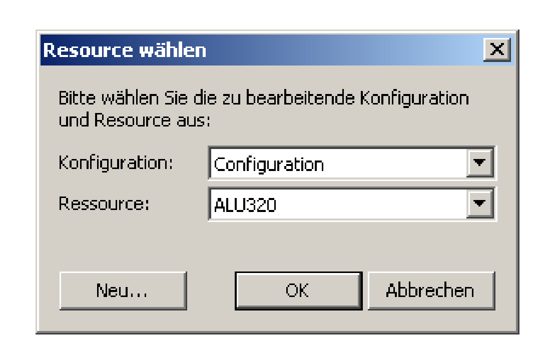
Sofern Sie Multiprog von der OHP Installations-CD inklusiver der OHP Add-Ons installiert haben, steht Ihnen der neue SPS-Konfigurator zur Verfügung. Sollte dies nicht der Fall sein, können sie mit der Beschreibung in Teil II, Kapitel 4.2 fortfahren.

Mit Hilfe des neuen SPS-Konfigurators kann die E/A-Konfiguration einer Anlage vom Anwender schnell und komfortabel durchgeführt werden.



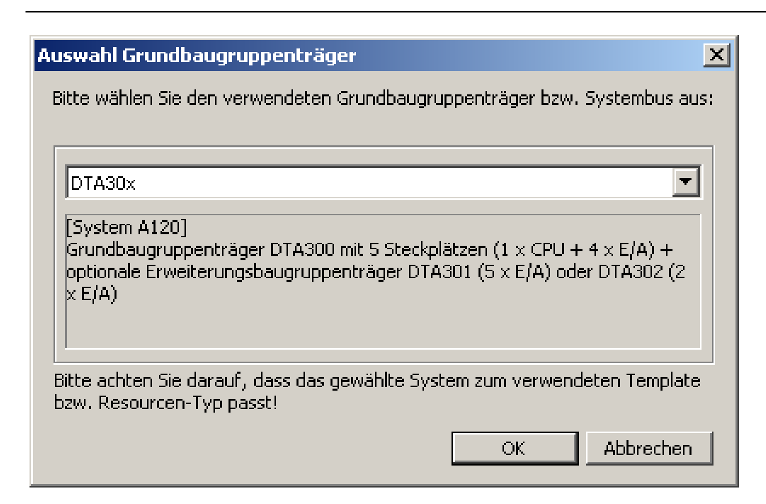
Ein Klick auf den Anwahlbutton "SPS-Konfigurator" in der Menüleiste öffnet den Dialog zur Auswahl der zu bearbeitenden SPS. Da eine SPS im Multiprog-Projektbaum durch einen Konfigurationsknoten und einen untergeordneten Ressourcen-Knoten repräsentiert wird, erfolgt die Auswahl über die entsprechenden Knotenbezeichnungen.





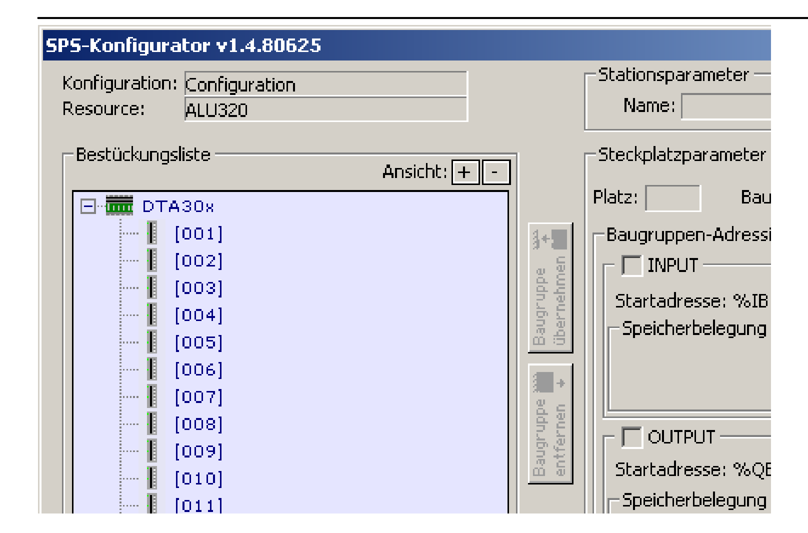
Nach Auswahl der zu bearbeitenden Konfiguration und Ressource und Klick auf Button "OK" wird eine bestehende Bestückungsliste geöffnet. Falls noch keine Bestückungsliste existiert, dann wird der Dialog "Auswahl Grundbaugruppenträger" aufgerufen:



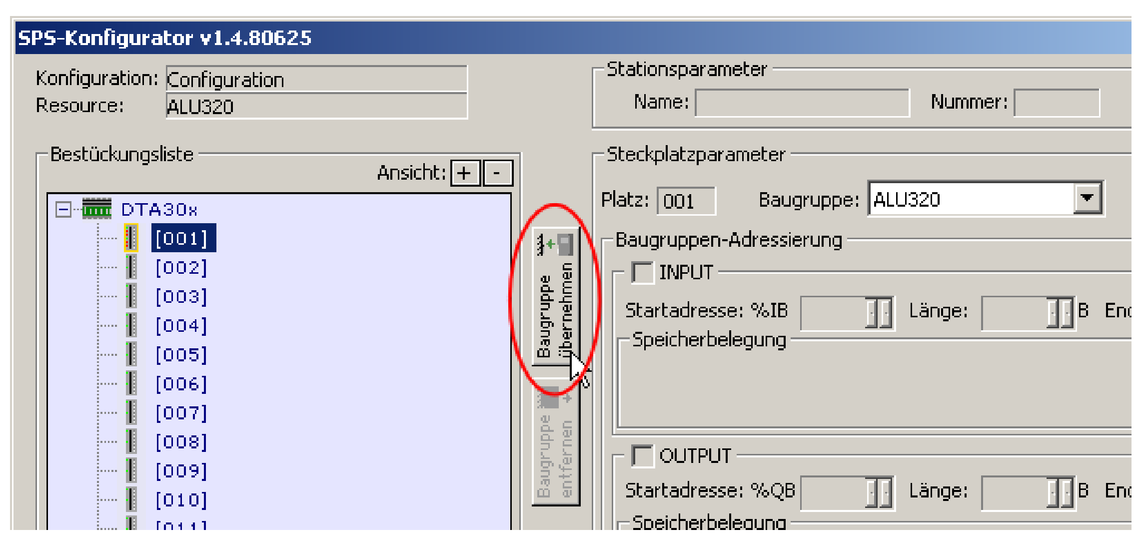


Dort wird für die Ressource ALU320 automatisch ein DTA30x Grundbaugruppenträger angeboten und nach Klick auf Button "OK" in die Bestückungsliste eingetragen:





Auf dem ersten Steckplatz (SP:001) kann im DTA300 nur eine ALU320 gesteckt werden, die schon automatisch im Baugruppen-Feld eingetragen wird.

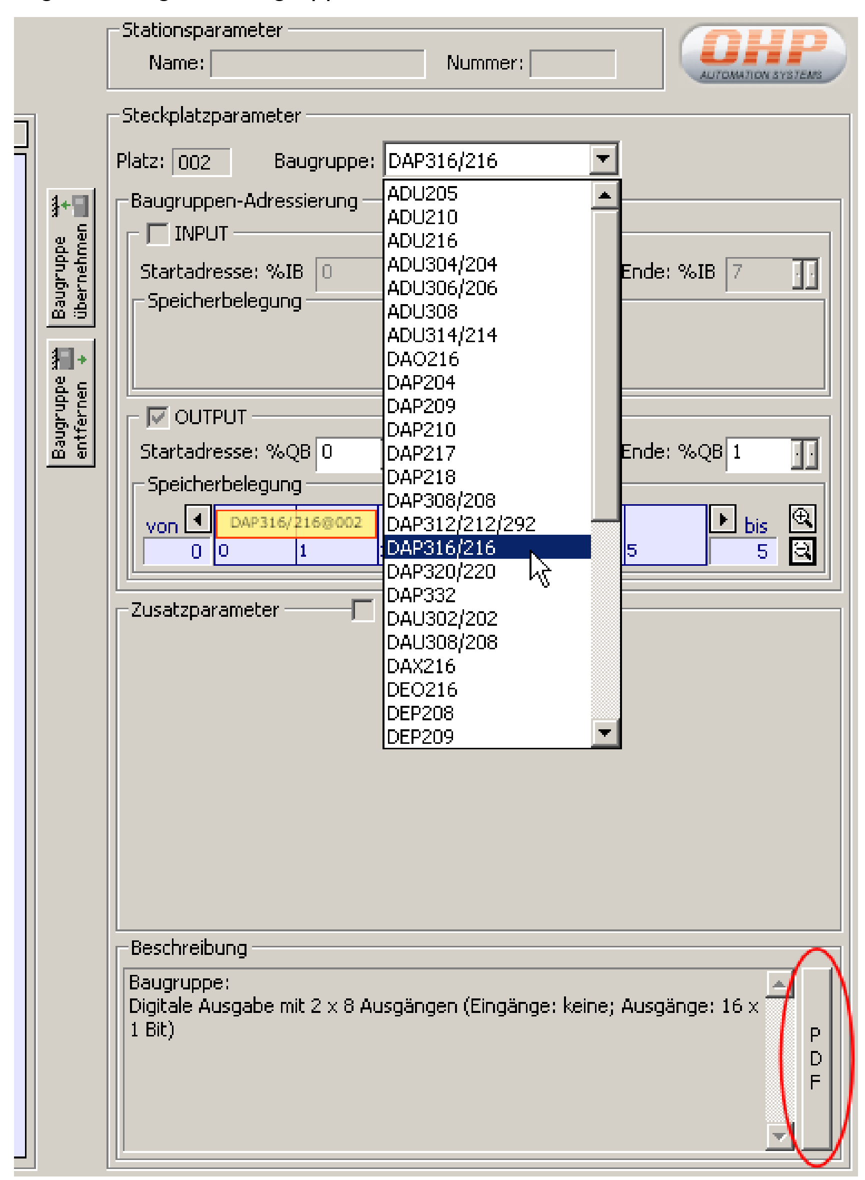


Nach einem Klick auf den Button "Baugruppe übernehmen" wird die ALU320 auf dem ersten Steckplatz [001] eingetragen und der nächste Steckplatz kann mit der Maus angewählt werden.



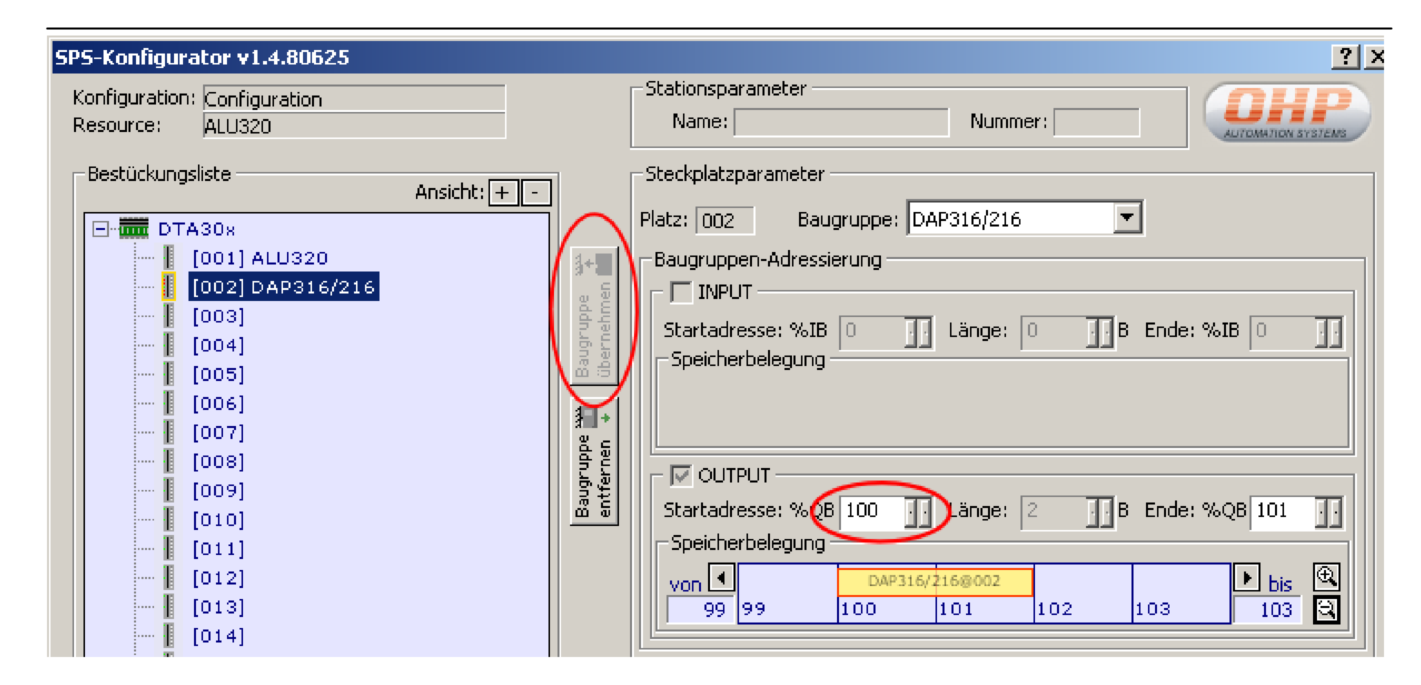
4.1.1 Definition Standard I/O Module

Digitale Ausgabe-Baugruppe:



Die gewünschte Baugruppe muss ausgewählt werden (hier eine digitale Ausgabe Karte mit 2 * 8 Ausgängen vom Typ DAP216 oder DAP316). Im Beschreibungsfeld wird eine Kurzanleitung zur jeweils angewählten Baugruppe angezeigt. Mit einem Klick auf den Button "PDF" kann eine detaillierte Beschreibung der OHP Baugruppe als PDF- Datei geladen werden, dazu muss das Programm "Adobe Reader" (mindestens in der Version 5.0) auf dem Programmiergerät installiert sein.





Der Speicherbereich der Ausgabebaugruppe muss in dem Feld "Startadresse" festgelegt werden, durch das Eintragen einer 100 starten in diesem Beispiel die benötigten 2 Bytes (2 * 8 Bits) der Ausgangs-Karte ab Ausgangs-Byte 100 (%QB100) und gehen bis Ausgangs-Byte 101 (%QB101). Der belegte Speicherbereich wird auch im Anzeigefeld "Speicherbelegung" als gelber Balken über den benutzten Byte-Adressen markiert.

Ein Klick auf den Button "Baugruppe übernehmen" trägt die Baugruppe auf den Steckplatz 2 in den Baugruppenträger ein.



Hinweis: Eine nachträgliche Veränderung z.B. der Startadresse, muss immer mit dem Button "Baugruppe übernehmen" bestätigt werden, bevor sich die Veränderung bemerkbar macht.



Analoge Eingabe-Baugruppe:

SPS-Konfigurator v1.4.80625

Konfiguration: Configuration
Resource: ALU320

Stationsparameter
Name: Nummer:

Bestückungsliste
Ansicht: + -

DTA30x
[001] ALU320
[002] DAP316/216
[003] ADU306/206
[004]
[005]
[006]
[007]
[008]
[009]
[010]
[011]
[012]
[013]
[014]
[015]
[016]
[017]
[018]
[019]
[020]

Baugruppe übernehmen
Baugruppe entfernen

Steckplatzparameter
Platz: 003 Baugruppe: ADU306/206

Baugruppen-Adressierung
INPUT
Startadresse: %IB 50 Länge: 8 B Ende: %IB 57
Speicherbelegung
von 46 46 48 ADU306/206@003 50 52 54 56 58 60 62 bis 62

OUTPUT
Startadresse: %QB 102 Länge: 2 B Ende: %QB 103
Speicherbelegung

Zusatzparameter dynamisch zur Laufzeit
Eingänge 1-4
Betriebsart: bipolar, unipolar, Drahtbr.-Überw., Überst.-Überw.
Messbereich: 1: ±10V 2: ±10V 3: ±1V/±20mA 4: ±1V/±20mA
Eingänge 5-8
Betriebsart: bipolar, unipolar, Drahtbr.-Überw., Überst.-Überw.
Messbereich: 5: ±1V/±20mA 6: ±1V/±20mA 7: ±1V/±20mA 8: ±1V/±20mA

Beschreibung
Baugruppe:
Eingabebaugruppe mit 4 analogen, potentialgetrennten Eingängen
(Eingänge: 4 x 16 Bit; Ausgänge: 1 Konfigurationsbytes)
Konfiguration der Baugruppe:
Eingänge 1-4:
Betriebsart: Bipolar, ohne Drahtbruchüberwachung, mit Übersteuerungsüberwachung

PDF

ADU306

Das Beispiel zeigt die Bestückung des Steckplatzes 3 mit einer analogen Eingabe-Baugruppe ADU306, mit 4 analogen, potenzialgetrennten Eingängen. Die benötigten 8 Bytes (4 * 16 Bits) beginnen ab Eingangs-Byte 50 (%IB50) und belegen bis Eingangs-Byte 57 (%IB57) den Eingangs-Speicher. Die Betriebsart für alle 4 Kanäle ist bipolar, mit Kanal 1 und 2 im ±10V Messbereich und Kanal 3 und 4 im ± 1V/±20mA Messbereich



ADU308

Hier wird im Beispiel die Anwahl von Steckplatz 4 mit einer analogen Eingabe-Baugruppe mit 8 analogen, potentialgetrennten Eingängen dargestellt.

Das System reserviert die benötigten 16 Byte (8 * 16 Bits) im Anschluss an den bereits durch den auf Steckplatz 3 eingetragenen ADU306 belegten Eingangs-Speicherbereich von Eingangs-Byte 58 (%IB58) bis Eingangs-Byte 73 (%IB73). Dargestellt wird dies durch den gelben Balken mit der Beschriftung "Vorschau", der orange Balken mit der Beschriftung "ADU306@003" zeigt den bereits belegten Speicherbereich an.

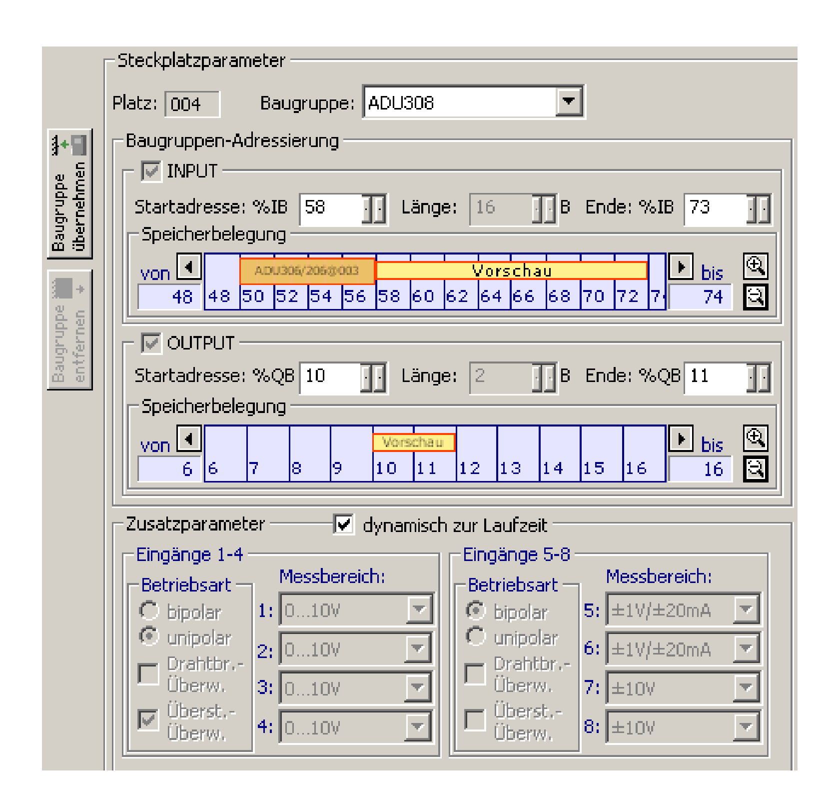
Die Kanäle 1 bis 4 sind unipolar auf den Messbereich 0…10V und die Kanäle 5 bis 8 sind bipolar auf den Messbereich ±1V/±20mA (Kanal 5 und 6) und ±10V (Kanal 7 und 8) eingestellt.



Hinweis: Wahlweise können für den ADU306 ein Ausgabebyte und für den ADU308 zwei Ausgabebyte (%QB) projektiert werden. Der Messbereich kann dann über diese Byte [hier Ausgabe-Byte 10 (%QB10)



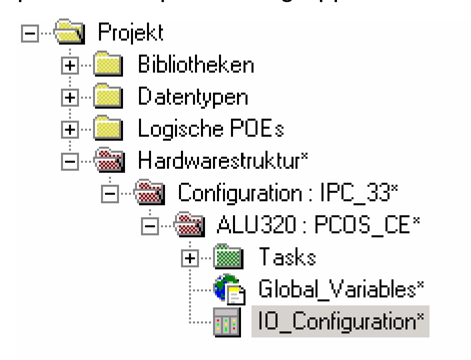
und Ausgabe-Byte 11 (%QB11)] genauso wie über Parameter 3 bzw. 4 eingestellt werden (siehe dazu auch die Beispiele aus Teil II, Kapitel "4.2.1). Über die Ausgabebyte kann der Messbereich durch das Anwenderprogramm dynamisch geändert werden. Bei Einstellung über die Parameter, muss die SPS neu geladen werden..



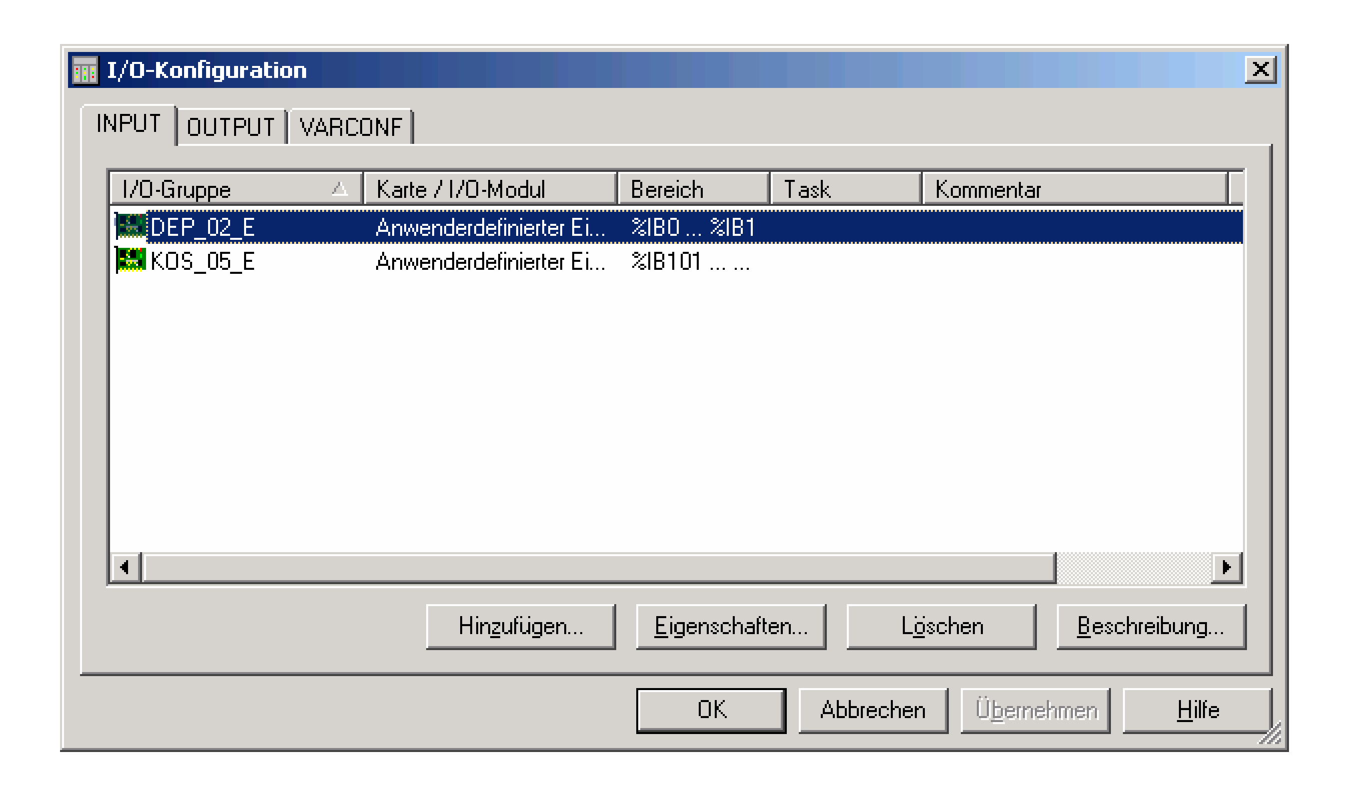


4.2 Ein- und Ausgaben mit der Multiprog IO_Configuration

Über die I/O-Konfiguration (Menü IO_Configuration) werden die Schnittstellen zu den Ein- und Ausgabebaugruppen, den Expertenbaugruppen und der ModbusTCP Kopplung definiert.



Die IO_Configuration ist ein Untermenü in der Hardwarestruktur und kann durch Doppelklick oder über das Pop-Up-Menü (Anwahl über die rechte Maustaste) geöffnet werden.



Achtung:

Der Name muss für jede Baugruppe eindeutig angegeben werden. Dies gilt auch bei gemischten Baugruppen wie der DAP 212 oder der KOS xxx. Hier muss jeweils ein Eintrag im Register INPUT und OUTPUT erfolgen, die Namen dürfen nicht identisch sein. Es können maximal 8 Zeichen eingegeben werden.

Beispiel: DEP_02_E für die Eingänge der DEP 316 auf Steckplatz 2
KOS_05_E und KOS_05_A für die Ein- und Ausgabebyte der KOS auf Steckplatz 5.

Die Längenangabe für die logischen Adressen ist baugruppenabhängig und können der folgenden Tabelle entnommen werden.

Baugruppenübersicht

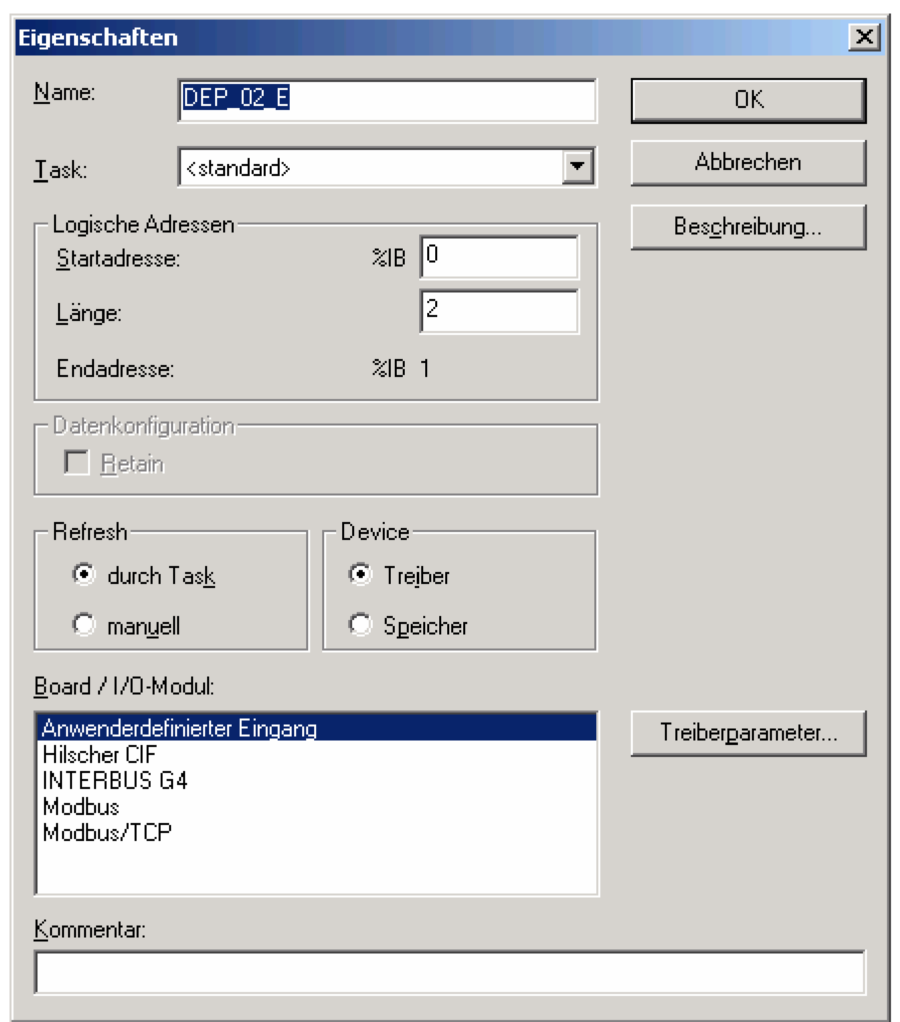
Baugruppe	Ident-code	Anzahl Byte Eingabe	Ausgabe	Bemerkung
				Gemischte digital Ein- und Ausgabe
DAP212 / DAP292 / DAP 312	8	1	1	8 E + 4 A
DAP220 / DAP 320	9	1	1	8 E + 8 A
				Digital Eingabe
DEP216/DEP316 DEP296/DEP297	12	2		2 x 8 E
DEO216	13	2		2 x 8 E
DEP220	14	2		2 x 8 E
DEP208	15	1		1 x 8 E
DEP210	16	1		1 x 8 E
DEP211	17	1		1 x 8 E
DEP218	18	2		2 x 8 E
DEP209	19	1		1 x 8 E
DEP214 / DEP 314	20	2		2 x 8 E
DEX216	44	2		
DEP217	45	2		2 x 8 E
DEP215	46	2		2 x 8 E
DEP332	70	4		4 x 8 E
				Digital Ausgabe
DAP204	23		1	4 A
DAP208 / DAP 308	24		1	8 A
DAP216 / DAP 316	25		2	2 x 8 A
DAP209	26		1	8 A
DAO216	27		2	2 x 8 A
DAP210	28		1	8 A
DAP218	29		2	2 x 8 A
DAX216	41		2	2 x 8 A
DAP217	42		2	2 x 8 A
DAP332	71		4	4 X 8 A in Vorbereitung



				Analog Eingabe (ADU)
ADU204 / ADU 304	32	8		4 x AE
ADU205	34	8		4 x AE
ADU206 / ADU 306	35	8	1 *)	4 x AE
ADU308	38	16	2 *)	8 x AE
ADU 214 / ADU 314	36	16	8 *)	8 x AE
ADU 210	56	8	2 *)	4 x AE
ADU 216	59	8		4 x AE
				Analog Ausgabe (DAU)
DAU202 / DAU 302	48		4	2 x AA
DAU208 / DAU 308	50		16	8 x AA
				Motorsteuerkarte
MSK 351	80	4	4	16 x DI, 12 x DA, 1 x AE
				Kommunikationsbaugruppen
KOS201 / KOS202A	90	128	128	Für Unterstationen und Zentralen
KOS201 / KOS202A	92	32	32	Für Unterzentralen
KOS 203	93	128	128	Für Unterstationen
KOS 260	91	128	128	Für Hauptbaugruppenträger
KOS 260	95	48	48	Für Erweiterungsbaugruppenträger

*) Die Ausgabebyte (%QB) können zur Parametrierung verwendet werden.

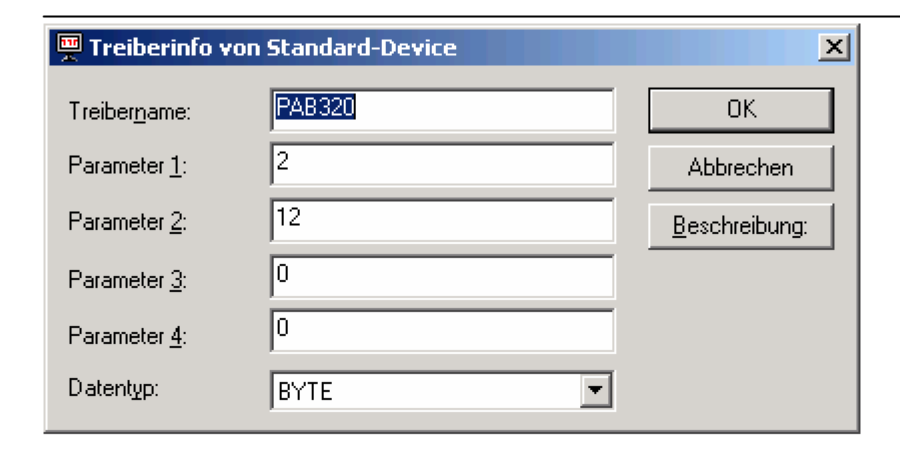
4.2.1 Definition Standard I/O Module



Als Board / I/O-Modul ist immer die Einstellung "Anwenderdefinierter Eingang/Ausgang" zu wählen.

Rufen Sie anschließend über den Button "Treiberparameter…" das Menü zur Eingabe der Treiberinfo auf.

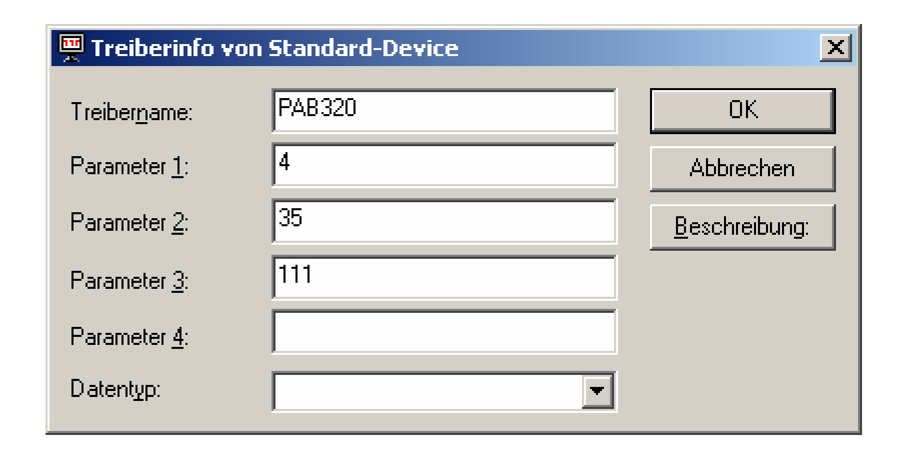




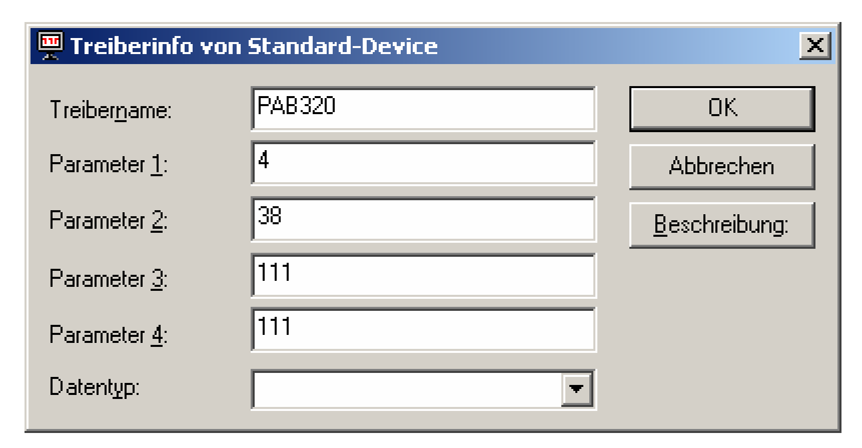
Eingabe **Treiberparameter**

- Als Treibername ist immer **PAB320** einzutragen.
- In Parameter 1 ist die Steckplatzadresse der Baugruppe einzutragen.
- In Parameter 2 wird der Identcode der Baugruppe eingetragen.
- Parameter 3 und 4 enthalten bei einigen Modulen baugruppenspezifische Parameter. Z.B wird für einige ADUs die Messbereichsauswahl als Dezimalwert vorgegeben.

Messbereichsauswahl ADUs



ADU 206/306



ADU 308

Die zweite 4er Gruppe von Messwerten wird über den Parameter 4 definiert.

Die Grundstellung 0 bedeutet:

ADU 308

Alle 4 Eingänge auf Messbereich ±1 V bzw. ±20 mA je nach Verdrahtung der Anschlüsse bzw. bei je nach Einstellung der DIL-Schalter.

Keine Überwachung bei Übersteuerung

Keine Überwachung auf Drahtbruch

Bipolarer Betrieb

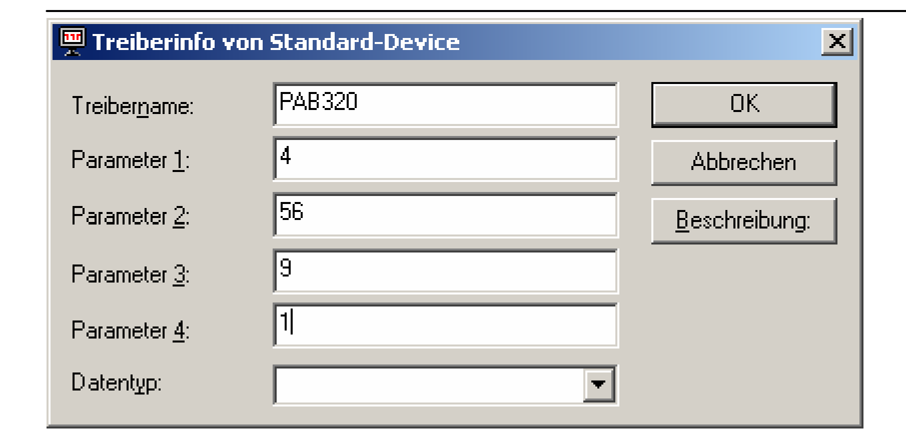
Abweichend von der Grundeinstellung sind folgende weitere Voreinstellungen individuell möglich. Diese Einstellung ergibt sich durch die Addition der Werte:

1	(Bit 2^0)	Eingang 1 auf Messbereich ±10 V
2	(Bit 2^1)	Eingang 2 auf Messbereich ±10 V
4	(Bit 2^2)	Eingang 3 auf Messbereich ±10 V
8	(Bit 2^3)	Eingang 4 auf Messbereich ±10 V
16	(Bit 2^4)	Unipolarer Betrieb, Auflösung 12 Bit ohne Vorzeichen, nicht kombinierbar mit Drahtbruchüberwachung, bei Ausgabe Umrechnung der Digitalwerte erforderlich.
32	(Bit 2^5)	Alle 4 Eingänge auf Messereich 4 ... 20 mA bei Brückenverwendung an den Eingängen mit gleichzeitiger Überwachung auf Drahtbruch bei Strömen < 2.08 mA
47	(Bit 2^0 - 2^3,2^5)	oder alle 4 Eingänge auf Messbereich 2 ... 10 V, keine Brücken an den Eingängen
64	(Bit 2^6)	Überwachung auf Messwerte größer Nennwert + Toleranz (Übersteuerung) an allen 4 Eingängen.
128	(Bit 2^7)	Ohne Bedeutung, Einstellung bleibt 0



Hinweis: Wahlweise können für den ADU 306 ein und für den ADU 308 zwei Ausgabebyte (%QB) projektiert werden. Der Messbereich kann dann über dies Byte genauso wie über Parameter 3 bzw. 4 eingestellt werden. Über die Ausgabebyte kann der Messbereich durch das Anwenderprogramm dynamisch geändert werden. Bei Einstellung über die Parameter, muss die SPS neu geladen werden.





ADU 210

Es werden jeweils 2 Kanäle gemeinsam festgelegt.

Achtung: Abweichend zur Projektierung mit AKF oder Concept können mit Multiprog die Kanäle derzeit nicht einzeln parametriert werden.

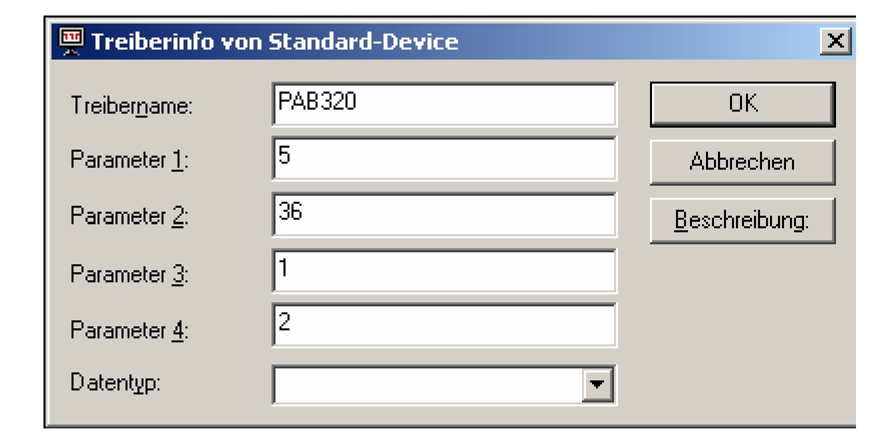
Kanal 1 und 2 werden gemeinsam über Parameter 3 und Kanal 3 und 4 über Parameter 4 eingestellt.

Im Beispiel sind die Kanäle 1 und 2 auf 2…10 V und Grenzwert 0 eingestellt. Kanal 3 und 4 sind auf Messbereich 0…10 V und Grenzwert 0 eingestellt.

Die Kennzahlen für die Einstellung der Messbereich entnehmen Sie bitte der Baugruppenbeschreibungen des ADUs.



Hinweis: Wahlweise kann der ADU 210 über zwei Ausgabebyte (%QB) projektiert werden. Der Messbereich kann dann über dies Byte genauso wie über Parameter 3 bzw. 4 eingestellt werden. Über die Ausgabebyte kann der Messbereich durch das Anwenderprogramm dynamisch geändert werden. Bei Einstellung über die Parameter, muss die SPS neu geladen werden.



Beim ADU 214 / ADU 314 werden die Kanäle 1 bis 4 über Parameter 3 und die Kanäle 5 bis 8 über Parameter 4 eingestellt. Im Beispiel sind die Kanäle 1 bis 4 auf 0…10 V und Kanal 5 bis 8 sind auf Messbereich 0…5 V eingestellt.

Die Kennzahlen für die Einstellung der Messbereich entnehmen sie bitte die jeweiligen Baugruppenbeschreibungen der ADUs.

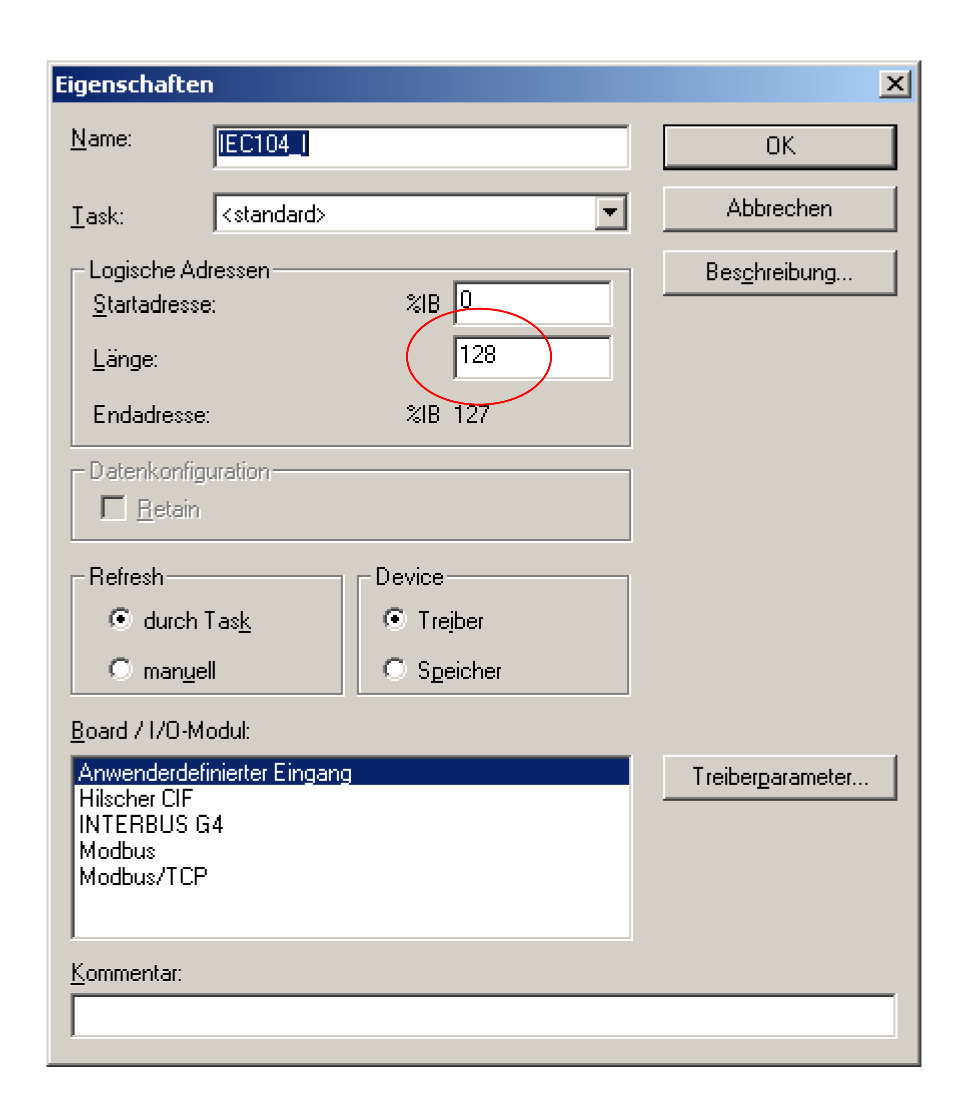


Hinweis: Wahlweise kann jeder einzelne Eingang des ADU 214 / 314 über 8 Ausgabebyte (%QB) projektiert werden. Über die Ausgabebyte kann der Messbereich durch das Anwenderprogramm dynamisch geändert werden. Bei Einstellung über die Parameter, muss die SPS neu geladen werden.



4.3 I/O Konfiguration für IEC 60870-5-101 oder IEC 60870-5-104

Die Bereiche für die Übergabe von Daten an die Kommunikationsschnittstelle der AU320 werden wie eine virtuelle I/O Baugruppe definiert. Es ist möglich mehrere dieser virtuellen Baugruppen zu projektieren.



Länge für Input

max. 512

Länge für Output
max. 1024

Der Name kann vom Anwender frei vergeben werden. Er kann aus bis zu 8 Zeichen bestehen und muss eindeutig sein. Es empfiehlt sich die Ein- und Ausgaberichtung durch ein I / O oder E / A zu kennzeichnen.

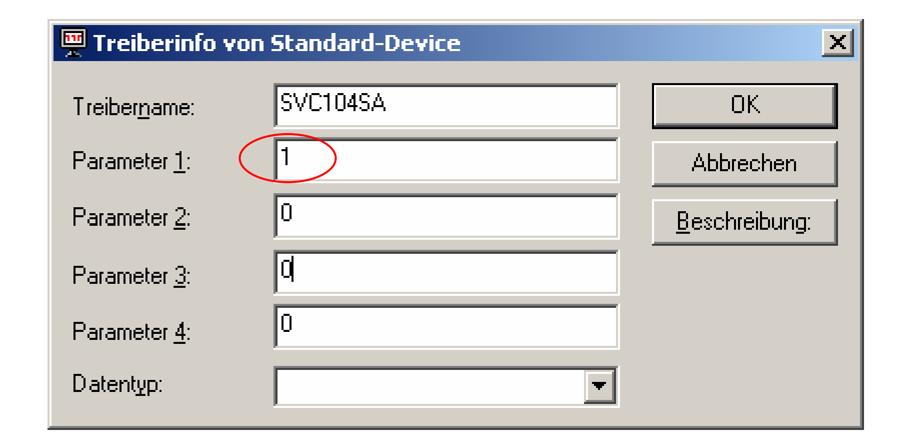
Anschließend werden über die **Treiberparameter** des **Anwenderdefinierten Eingangs/Ausgangs** die Service-Routinen für IEC 60870-5-101 oder -104 zugeordnet.

Die ALU 320 kann bei entsprechender Projektierung (Datei PCOS_A320.INI) gleichzeitig mehrere Service-Routinen bedienen.

Beispiel:

Service1 = IEC 60870-5-104 (Treiber = SVC104SA, TCP/IP Schnittstelle)
Service2 = IEC 60870-5-101 (Treiber = SVC101SA, serielle Schnittstelle COM 1)
Service3 = IEC 60870-5-101 (Treiber = SVC101SA, serielle Schnittstelle COM 2)

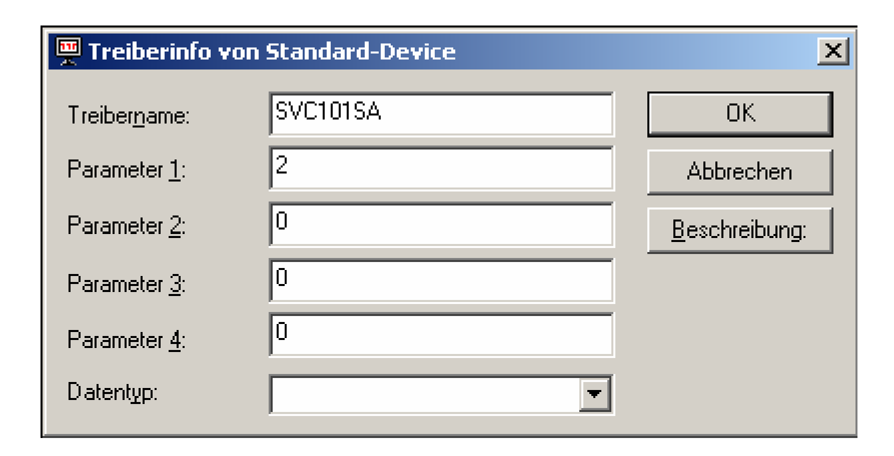




SVC104SA ist der Service für das IEC 60870-5-104 Protokoll.

Service Nummer in Parameter 1 muss identisch mit den Nummern in den Dateien SVC_COMn.INI und SVC_RTUn.INI sein.

Siehe auch Kap. 6



SVC101SA ist der Service für das IEC 60870-5-101 Protokoll.

Service Nummer in Parameter 1 muss identisch mit den Nummern in den Dateien SVC_COMn.INI und SVC_RTUn.INI sein.

Siehe auch Kap. 6

Achtung:

Achten Sie unbedingt darauf, dass die in der I/O Konfiguration projektierten Service-Routinen in der INI-Datei PCOS_A320.INI freigegeben sind, bevor Sie das Projekt an die ALU 320 senden. Wenn diese Routinen nicht freigegeben sind, wird der Ladevorgang mit Fehler abgebrochen.

Weiterführende Informationen siehe Kapitel 0 und Kapitel 6.4.



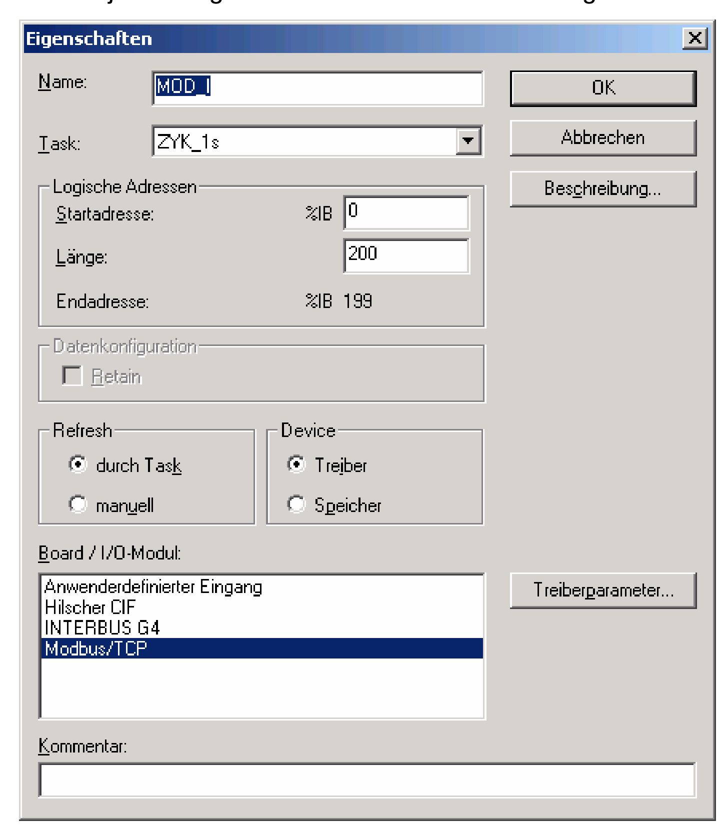
Hinweis: Ab Firmware-Version 1.70 werden geänderte Dateinamen verwendet (PCOSA320.INI, SVCCOMn.INI und SVCRTUn.INI)



4.4 I/O Konfiguration für Modbus TCP Client

Der Treiber wird im Auswahlfenster "Board / I/O-Modul" angezeigt, wenn Sie in der I/O-Konfiguration eine neue Gruppe hinzufügen.

Die Projektierung wird anhand des INPUT Dialogs der I/O-Konfiguration erläutert.



Task:

Um feste Abfrageintervalle für den Modbus zu erreichen, definieren Sie für den Aufruf des Treibers eine zyklische Task. Falls in Ihrer Applikation keine zyklischen Programmaufrufe benötigt werden, können sie hierfür auch ein leeres Dummy-Programm erstellen.

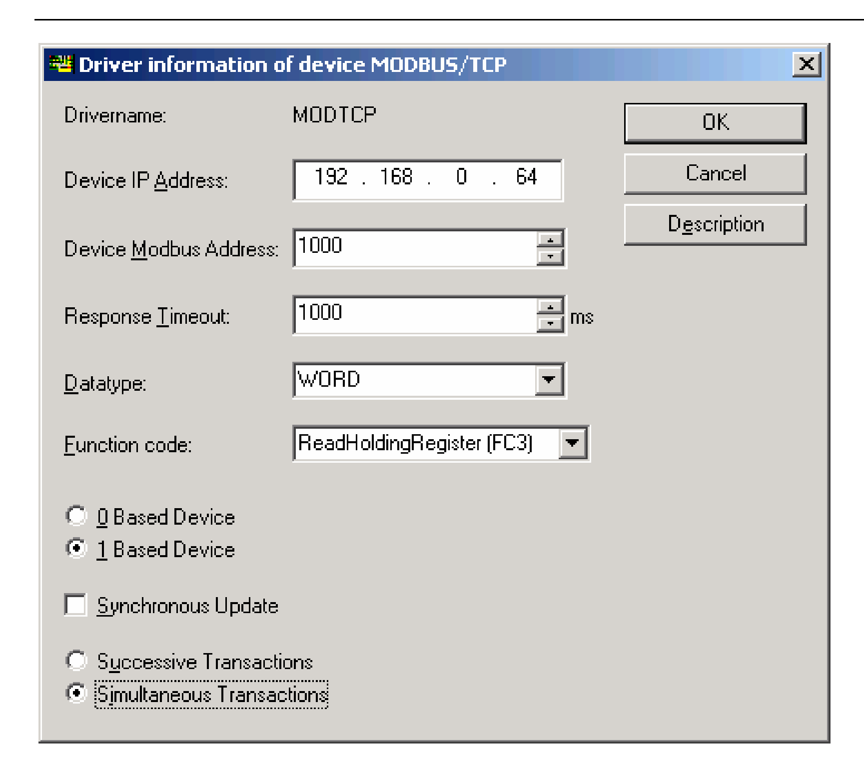
Logische Adressen:

In diesem Abfragedialog geben Sie das 1. IB an, ab dem die übertragenen Daten abgelegt werden.

Länge:

Die Länge der abgefragten Daten wird in Anzahl Byte angegeben. Im obigen Beispiel werden 200 Byte = 100 WORD bzw. Register aus der angekoppelten Station abgefragt.

Über den Dialog "Treiberparameter…" wird festgelegt, welche Register aus der Station ausgelesen werden.



Device IP Adresse:

In das Feld Device IP Adresse tragen Sie die IP-Adresse der angekoppelten Station ein.

Device Modbus Address:

In das Feld Device Modbus Address tragen Sie die Adresse des 1. Registers ein, ab dem ausgelesen werden soll.

In der Beispiel-Projektierung werden ab Register 1000 (401000, MW 1000 etc.) 100 Register aus der angekoppelten Station abgefragt und in Multiprog im Bereich IB0 … IB199 abgelegt.

Funktion code:

Hier können Sie einstellen, ob Sie auf Register (WORD), Coils (BIT) oder direkt auf die I/O der angeschlossenen Station zugreifen wollen.

Der ausgewählte Funktionscode muss durch die angeschlossene Station unterstützt werden.

Mit **Based Device** 0 oder 1 geben Sie an, ob die angekoppelte Station in ihrer Zählweise mit Register 0 oder 1 beginnt.

Über die Definition mehrere I/O-Gruppen können Sie verschiedene Bereiche aus der angekoppelten Station abfragen. Die Bereiche können nacheinander (**Successive Transaction**) oder parallel (**Simultaneouse Transaction**) abgefragt werden.

Nacheinender heißt, dass das Abfragetelegramm für den nächsten Bereich erst dann abgeschickt wird, wenn die Antwort auf die zuvor gesendete Abfrage eingetroffen ist, oder die unter **Response Timeout** projektierte Zeit abgelaufen ist.

Parallel heißt, dass mehrer Abfragetelegramme hintereinander geschickt werden können, unabhängig davon, ob eine Antwort eingetroffen ist. Die Datenübertragung ist also asynchron.

Für die Projektierung von Ausgaben über den Modbus/TCP gehen Sie in der gleichen Weise vor wie für die Eingaben. Benutzen Sie hierfür den OUTPUT Dialog der I/O-Konfiguration.



4.5 Projektierung des Shared-Memory-Bereichs für Modbus RS232-Slave / TCP-Server

Die ALU 320 stellt insgesamt einen Bereich von 16 kB (MB3.0000 bis MB3.16383) für das Shared-Memory zur Verfügung. Dieser Bereich oder Teile davon, können für die Modbus RS232-Slave oder TCP-Server Schnittstelle reserviert werden.

Die Definition des Bereichs erfolgt über INI-Dateien, die auf der CF-Karte abgelegt werden. Diese Dateien PCOS_A320.INI und MODBUS_SERVER.INI befinden sich im Verzeichnis PCOS_OHP.

 Weiterführende Informationen siehe Kap. 6.5.



Hinweis: Ab Firmware-Version 1.70 werden geänderte Dateinamen verwendet (PCOSA320.INI und MBSRV.INI)

5 Systemmerker

Für die maximal 20 Steckplätze einer @120 (DTA300 + 3 x DTA 301) sind im Systemmerkerbereich mehrere Doppelwort für die Fehler- und Statusanzeige definiert. Statusinformationen werden nur für Steckplätze geliefert, die in der I/O-Konfiguration definiert sind.

Bei einfachen E/A-Bauguppen werden nur die Systemmerker "Baugruppenausfall" und "Baugruppenstörung" bedient. Für Expertenbaugruppen (z.B. KOS xxx) und komplexere E/A-Module (z.B. ADU 206) werden zusätzlich noch 4 Statusbyte zur Verfügung gestellt.

Der Systemmerkerbereich hat eine feste Struktur. Merkerbit und Merkerbyte für Steckplätze die nicht vorhanden bzw. nicht bestückt sind, sind immer 0.



Hinweis für Programmierer die A120-Systeme umrüsten auf @120.
Das Systemmerkerdoppelwort "Baugruppenausfall" %MD1.3000 entspricht den SM31 … SM48 in AKF12.
Das Systemmerkerdoppelwort "Baugruppenstörung" %MD1.3004 entspricht den SM2.1, SM3.1 etc. in AKF12.

Die Systemmerker können als Doppelwort, Byte oder Bit angesprochen werden.

Um einen Sammelfehler für Baugruppenausfall zu erzeugen kann z.B. das %MD1.3000 auf ungleich NULL geprüft werden. Für die Prüfung einzelner Steckplätze verwendet man sinnvoller weise die Merkerbit %MX1.3000.1, %MX1.3000.2 usw.

Die Definition entnehmen Sie bitte den folgenden Tabellen:



5.1 Systemmerker Service

%MD1.2952	%MB1.2952 %MB1.2953 %MB1.2954 %MB1.2955	Statusbyte DCF-Empfang (in Vorbereitung)
%MD1.2956	%MB1.2956 %MB1.2957 %MB1.2958 %MB1.2959	Statusbyte Service Sammelmerker Statusbyte Service Sammelmerker Statusbyte Service Sammelmerker Statusbyte Service Sammelmerker
%MD1.2960	%MB1.2960 %MB1.2961 %MB1.2962 %MB1.2963	Statusbyte1 Service 1 Statusbyte2 Service 1 Statusbyte3 Service 1 Statusbyte4 Service 1
. . .	. . .	. . .
%MD1.2992	%MB1.2992 %MB1.2993 %MB1.2994 %MB1.2995	Statusbyte1 Service 9 Statusbyte2 Service 9 Statusbyte3 Service 9 Statusbyte4 Service 9

Statusbyte Service Sammelmerker MD1.2956:

Bitposition entspricht Sammelfehler eines System-Service SVC0 bis SVC9 (SVC0 ist immer für internen System-Service reserviert).

SVC1 bis SVC 9 können derzeit für IEC 60870-5-101 oder IEC 60870-5-104 Kommunikation projektiert werden.

Sammelfehler-Bit0 von MD1.2956 reserviert
Sammelfehler-Bit1 von MD1.2956 für Service 1
Sammelfehler-Bit2 von MD1.2956 für Service 2
usw.

Statusbyte1 Service n:

Bit 0	nicht definiert
Bit 1	1 = Teilnehmer hat keine Verbindung, 0 = Verbindung OK (Zykl. Timeout)
Bit 2	1 = Bus keine Verbindung, 0 = Bus Ok (Bus Timeout)
Bit 3…7	nicht definiert

Die Statusbyte 2…4 werden derzeit nicht verwendet.

5.2 Systemmerker Steckplatz in I/O Konfiguration belegt

%MD1.2996	%MB1.2996	%MX1.2996.0	ALU (Platz 1) - reserviert	
		%MX1.2996.1	E/A Karte (Platz 2)	Merker für Platz 1 bis 20
		%MX1.2996.2	E/A Karte (Platz 3)	
		%MX1.2996.3	E/A Karte (Platz 4)	0 … nicht projektiert
		%MX1.2996.4	E/A Karte (Platz 5)	1 … belegt
		%MX1.2996.5	E/A Karte (Platz 6)	
		%MX1.2996.6	E/A Karte (Platz 7)	
		%MX1.2996.7	E/A Karte (Platz 8)	
	%MB1.2997	%MX1.2997.0	E/A Karte (Platz 9)	
		%MX1.2997.1	E/A Karte (Platz 10)	
		%MX1.2997.2	E/A Karte (Platz 11)	
		%MX1.2997.3	E/A Karte (Platz 12)	
		%MX1.2997.4	E/A Karte (Platz 13)	
		%MX1.2997.5	E/A Karte (Platz 14)	
		%MX1.2997.6	E/A Karte (Platz 15)	
		%MX1.2997.7	E/A Karte (Platz 16)	
	%MB1.2998	%MX1.2998.0	E/A Karte (Platz 17)	
		%MX1.2998.1	E/A Karte (Platz 18)	
		%MX1.2998.2	E/A Karte (Platz 19)	
		%MX1.2998.3	E/A Karte (Platz 20)	
		%MX1.2998.4	Frei (0)	
		.		
		.		
		%MX1.2998.7	Frei (0)	
	%MB1.2999	%MX1.2999.0	Reserviert	
		.		
		.		
		%MX1.2999.7		

Diese Merker zeigen an, ob ein Steckplatz in der I/O Konfiguration belegt wurde. Ist ein Platz projektiert, wird das entsprechende Bit auf 1 gesetzt.



5.3 Systemmerker Baugruppenausfall

<table>
<tr><td rowspan="28">%MD1.3000</td><td rowspan="8">%MB1.3000</td><td>%MX1.3000.0</td><td>ALU (Platz 1) - reserviert</td><td rowspan="28">Merker für Karte 1 bis 20

0 … OK
1 … Ausgefallen –
nur projektierte Karten</td></tr>
<tr><td>%MX1.3000.1</td><td>E/A Karte (Platz 2)</td></tr>
<tr><td>%MX1.3000.2</td><td>E/A Karte (Platz 3)</td></tr>
<tr><td>%MX1.3000.3</td><td>E/A Karte (Platz 4)</td></tr>
<tr><td>%MX1.3000.4</td><td>E/A Karte (Platz 5)</td></tr>
<tr><td>%MX1.3000.5</td><td>E/A Karte (Platz 6)</td></tr>
<tr><td>%MX1.3000.6</td><td>E/A Karte (Platz 7)</td></tr>
<tr><td>%MX1.3000.7</td><td>E/A Karte (Platz 8)</td></tr>
<tr><td rowspan="8">%MB1.3001</td><td>%MX1.3001.0</td><td>E/A Karte (Platz 9)</td></tr>
<tr><td>%MX1.3001.1</td><td>E/A Karte (Platz 10)</td></tr>
<tr><td>%MX1.3001.2</td><td>E/A Karte (Platz 11)</td></tr>
<tr><td>%MX1.3001.3</td><td>E/A Karte (Platz 12)</td></tr>
<tr><td>%MX1.3001.4</td><td>E/A Karte (Platz 13)</td></tr>
<tr><td>%MX1.3001.5</td><td>E/A Karte (Platz 14)</td></tr>
<tr><td>%MX1.3001.6</td><td>E/A Karte (Platz 15)</td></tr>
<tr><td>%MX1.3001.7</td><td>E/A Karte (Platz 16)</td></tr>
<tr><td rowspan="8">%MB1.3002</td><td>%MX1.3002.0</td><td>E/A Karte (Platz 17)</td></tr>
<tr><td>%MX1.3002.1</td><td>E/A Karte (Platz 18)</td></tr>
<tr><td>%MX1.3002.2</td><td>E/A Karte (Platz 19)</td></tr>
<tr><td>%MX1.3002.3</td><td>E/A Karte (Platz 20)</td></tr>
<tr><td>%MX1.3002.4</td><td>Frei (0)</td></tr>
<tr><td>.</td><td></td></tr>
<tr><td>.</td><td></td></tr>
<tr><td>%MX1.3002.7</td><td>Frei (0)</td></tr>
<tr><td rowspan="4">%MB1.3003</td><td>%MX1.3003.0</td><td>Reserviert</td></tr>
<tr><td>.</td><td></td></tr>
<tr><td>.</td><td></td></tr>
<tr><td>%MX1.3003.7</td><td></td></tr>
</table>

Ist ein Steckplatz in der I/O Konfiguration projektiert und nicht bestückt bzw. die Baugruppe defekt, wird diese als ausgefallen gemeldet.

5.4 Systemmerker Baugruppenstörung

%MD1.3004	%MB1.3004	%MX1.3004.0	ALU (Platz 1) - reserviert	
		%MX1.3004.1	E/A Karte (Platz 2)	Merker für Karte 1 bis 20
		%MX1.3004.2	E/A Karte (Platz 3)	
		%MX1.3004.3	E/A Karte (Platz 4)	0 … OK
		%MX1.3004.4	E/A Karte (Platz 5)	1 … **Störung**
		%MX1.3004.5	E/A Karte (Platz 6)	
		%MX1.3004.6	E/A Karte (Platz 7)	
		%MX1.3004.7	E/A Karte (Platz 8)	
	%MB1.3005	%MX1.3005.0	E/A Karte (Platz 9)	
		%MX1.3005.1	E/A Karte (Platz 10)	
		%MX1.3005.2	E/A Karte (Platz 11)	
		%MX1.3005.3	E/A Karte (Platz 12)	
		%MX1.3005.4	E/A Karte (Platz 13)	
		%MX1.3005.5	E/A Karte (Platz 14)	
		%MX1.3005.6	E/A Karte (Platz 15)	
		%MX1.3005.7	E/A Karte (Platz 16)	
	%MB1.3006	%MX1.3006.0	E/A Karte (Platz 17)	
		%MX1.3006.1	E/A Karte (Platz 18)	
		%MX1.3006.2	E/A Karte (Platz 19)	
		%MX1.3006.3	E/A Karte (Platz 20)	
		%MX1.3006.4	Frei (0)	
		.		
		.		
		%MX1.3006.7	Frei (0)	
	%MB1.3007	%MX1.3007.0	Reserviert	
		.		
		.		
		%MX1.3007.7		

Einige Baugruppen wie z.B. die KOSxxx oder ADU 206/306 usw. können Störungen melden. Der Systemmerker Baugruppenstörung dient als Sammelmerker für diese Störungen. Die genaue Störungsursache wird über die Statusbyte (siehe Kap. 5.5) bekannt gegeben.



5.5 Systemmerker Statusbyte

%MD1.3008	%MB1.3008	Statusbyte E/A Karte – Platz 1
	%MB1.3009	Statusbyte E/A Karte – Platz 1
	%MB1.3010	Statusbyte E/A Karte – Platz 1
	%MB1.3011	Statusbyte E/A Karte – Platz 1
%MD1.3012	%MB1.3012	Statusbyte E/A Karte – Platz 2
	%MB1.3013	Statusbyte E/A Karte – Platz 2
	%MB1.3014	Statusbyte E/A Karte – Platz 2
	%MB1.3015	Statusbyte E/A Karte – Platz 2
.	.	.
.	.	.
.	.	.
%MD1.3084	%MB1.3084	Statusbyte E/A Karte – Platz 20
	%MB1.3085	Statusbyte E/A Karte – Platz 20
	%MB1.3086	Statusbyte E/A Karte – Platz 20
	%MB1.3087	Statusbyte E/A Karte – Platz 20

5.5.1 Definition Statusbyte KOS201

1. Statusbyte

Bit 0	Steckplatzfehler KOS
Bit 1	SEAB Verkehr gestört AWD: Anz. Wahlversuche erfolglos
Bit 2	Minutenimpuls fehlt
Bit 3	Zeit fehlt
Bit 4	Keine Parameter
Bit 5	
Bit 6	Batteriefehler
Bit 7	Sammelfehler Bit 0 bis 4

2. Statusbyte – nicht verwendet

3. Statusbyte – nicht verwendet

4. Stausbyte

Bit 0	KOS- AUSGABE 1=BUSY, 0= OK
Bit 1	
Bit 2	
Bit 3	
Bit 4	KOS- EINGABE 1=BUSY, 0= OK
Bit 5	
Bit 6	
Bit 7	



5.5.2 Uhrzeitübergabe durch KOS 201/202

Die KOS übergibt in 8 aufeinander folgenden Byte die Uhrzeit, sofern diese synchronisiert ist. Die Synchronisation kann über SEAB-1F Zeittelegramme oder über DCF-Funksignal erfolgen. Diese Uhrzeitübergabe kann wiederum zum Synchronisieren der CPU-Uhr verwendet werden.

Die 8 Byte sind wie folgt definiert:

KOS im UST-Betrieb	KOS im UZ-Betrieb	Bezeichnung	Codierung
121	17	Jahr	BCD
122	18	Monat	BCD
123	19	Wochentag	BCD
124	20	Tag	BCD
125	21	Stunde	BCD
126	22	Minute	BCD
127	23	Sonderzeichen	1. Bit = Umschaltung auf Reserveante 2. Bit = Ankündigung SZ/WZ Umschaltung 3. Bit = Sommerzeit 4. Bit = Winterzeit 5. Bit = Schaltsekunde
128	24	Sekunde	BCD

Die Angabe der Byte ist immer relativ zum projektierten %IB Bereich in der IO-Konfiguration zu sehen.

Im Beispiel-Template entspricht dies auch dem Bereich %IB121 … %IB128.

5.5.3 Definition Statusbyte ADU206/306/308

1. Statusbyte

Bit 0	Übersteuerung bzw. Drahtbruch bei Strömen <2.08 mA am Eingang 1
Bit 1	Übersteuerung bzw. Drahtbruch bei Strömen <2.08 mA am Eingang 2
Bit 2	Übersteuerung bzw. Drahtbruch bei Strömen <2.08 mA am Eingang 3
Bit 3	Übersteuerung bzw. Drahtbruch bei Strömen <2.08 mA am Eingang 4
Bit 4	Betriebsart Unipolar
Bit 5	2 ... 10 V / 4 ... 20 mA an Eingängen 1 ... 4
Bit 6	1= Externe 24Volt Versorgung fehlt
Bit 7	ADU nicht betriebsbereit, Ursache: Übersteuerung oder Drahtbruch bei einem der 4 Eingänge oder Prozessorüberwachung hat angesprochen oder U = 24 V fehlt oder AS–BADU ist noch in Initialisierungsphase.

2. Stausbyte

Beim ADU 308 ist das 2. Statusbyte wie Statusbyte 1 definiert, es stellt jedoch den Zustand der Messwerteingänge 5…8 dar.

5.5.4 Definition Statusbyte ADU210/214

Um die Fehlerauswertung in Multiprog zu vereinfachen wird das vom ADU übergebene Fehlerbyte durch den I/O-Treiber auf 2 Statusbyte verteilt.

Das 1. Stausbyte des jeweiligen Slots enthält die Fehlernummer (Bit 2^0 – 2^3 des ADU Fehlerbytes).

Das 2. Stausbyte des jeweiligen Slots enthält die Eingangsnummer (Bit 2^4 – 2^7 des ADU Fehlerbytes).

Die Kodierung der Fehlernummer entnehmen Sie bitte die jeweiligen Baugruppenbeschreibungen der ADUs.

5.5.5 Definition Statusbyte Modbus

1. Statusbyte

Bit 0	Schnittstelle: 1 = Bereit, 0 = Fehler
Bit 1	Schnittstelle: 1 = Läuft, 0 = Stopp
Bit 2	Reserve
Bit 3	Reserve
Bit 4	COM-Task: 1 = Bereit, 0 = Fehler
Bit 5	Reserve
Bit 6	Reserve
Bit 7	Fest auf 1



2. Statusbyte

Bit 0	Kommunikationsstatus: 1 = Verbunden, 0 = nicht verbunden
Bit 1	Reserve
Bit 2	Reserve
Bit 3	Reserve
Bit 4	Reserve
Bit 5	Reserve
Bit 6	Reserve
Bit 7	Reserve

3. Statusbyte

Nummer der Funktion, die den letzten Fehler verursacht hat.

4. Statusbyte

Modbus Fehlercode

1	Unbekannter bzw. unzulässiger Funktionscode
2	Abgefragte Adresse / Bereich nicht bekannt
3	Illegaler Abfragewert (abhängig vom Abfragetyp)

5. Statusbyte

Fortlaufender Fehler-Zähler mit Umschlag bei 255.

6. Statusbyte

Fortlaufender Antwort-Zähler mit Umschlag bei 255.

Die Speicher-Adresse und Anzahl der Statusbyte wird in der Datei MODBUS_SERVER.INI eingestellt. Die Statusbyte 3 …6 sind optional.

Weiterführende Informationen siehe Kapitel 6.5.1.

6 Projektierung der Initialisierungsdateien

Die meisten der im Folgenden beschriebenen Initialisierungsdateien können über die OHP-Toolbar erzeugt und zur ALU übertragen werden. Die Übertragung erfolgt teilweise beim "Projekt senden" und teilweise über einen FTP-Transfer im jeweiligen Projektierungsmenü. Geänderte Einstellungen werden erst nach einem Neustart der ALU wirksam.

Das Bearbeiten der Dateien mit einem Texteditor ist ebenfalls möglich.

6.1 Einstellungen in der ALU320.INI

Projektierung über SPS-Konfigurator "ALU-Parameter –> Grundeinstellungen bearbeiten", Übertragung per FTP.

Im ROOT-Verzeichnis der CF-Karte befindet sich die Datei ALU320.INI. Diese Datei beinhaltet die Einstellungen bezüglich der IP-Adresse und diverser Dienste für den Zugriff auf die ALU320.

```
#ALU320:  (GEN)

[DEVICE]
RASVPN= FALSE

[IP]
ADDRESS             = 192.168.0.32
NETMASK             = 255.255.255.0
DHCP                = 0

[FTP]
ENABLE              = 1

[TELNET]
ENABLE              = 1

[TIME-GPS-NMEA]
ENABLE              = FALSE
PORT                = 2
BAUD                = 4800              # darf nicht geändert werden
REFRESH             = 60                #  in Minuten
```

6.1.1 Freigabe von Device-Treibern

Bei einer ALU320-017 mit integriertem GPRS-Modem kann über dieses Modem eine Verbindung zum Provider und falls erforderlich eine VPN-Tunnel aufgebaut werden. Hierfür ist der Eintrag RASVPN=TRUE zu setzen.

Ist RASVPN=TRUE sucht die ALU320 beim Hochlauf nach der Datei ALURAS.INI. Diese Datei beinhaltet die Einstellungen für die GPRS-Verbindung.



6.1.2 Einstellen der IP-Adresse auf der ALU320

ADRESSE und NETMASK müssen immer vorgegeben werden. Wenn Sie die ALU320 über ein Gateway betreiben möchten, können Sie die Datei wie folgt erweitern.

Beispiel:

ADDRESS= 192.168.101.101
NETMASK= 255.255.255.0
GATEWAY= 192.168.101.1

In einigen Fällen ist es erforderlich, einer Schnittstelle mehrere IPv4-Adressen durch *IP-Aliasing* zu zuweisen. IP-Aliasing wird verwendet, um einen Host aus verschiedenen Subnetzen erreichbar zu machen. Über den Eintrag [IP-ALIAS-IF1] kann eine zweite IP Adresse zugeordnet werden.

Beispiel:

[IP-ALIAS-IF1]
ENABLE= TRUE / FALSE ... ALIAS aktivieren
ADDRESS= 172.10.24.101
NETMASK= 255.255.255.0
GATEWAY= 172.10.24.1

6.1.3 Zugriff über FTP oder TELNET

Die Einträge [FTP] und [TELNET] ermöglichen den Zugriff auf die ALU mit den entsprechenden Diensten. Wenn Sie den Zugriff über diese Dienste nicht erlauben wollen, setzten Sie ENABLE =0.

Achtung: Wenn Sie den FTP-Zugriff deaktivieren, können Sie die INI-Dateien auf der ALU320 nur noch über einen Card-Reader ändern.

6.1.4 Zeitsynchronisation über GPS

Parameter unter [TIME-GPS-NMEA]

Wird der Parameter **ENABLE** auf TRUE gesetzt, kann die Zeitsynchronisation der ALU 320 über einen GPS-Empfänger erfolgen.

PORT gibt die COM-Schnittstelle (1 oder 2) an, an den der Empfänger angeschlossen wird.

BAUD gibt die Übertragungsgeschwindigkeit an, mit der der Empfänger die Daten an die ALU320 überträgt. Dieser Eintrag darf nicht verändert werden.

REFRESH gibt das Zeitintervall in Minuten (1…60) an, in dem die Systemuhr durch den GPS-Empfänger synchronisiert wird. D.h. innerhalb dieser Zeit treffen zwar mehrere Zeitinformation vom GPS-Empfänger ein, jedoch wird die Systemuhr erst nach Ablauf dieser Zeit durch das nächste gültige Zeittelegramm synchronisiert. Nach jeder erfolgreichen Synchronisation wird der Refresh-Timer wieder neu gestartet. Siehe auch Kapitel 8.1.

Achtung: Sofern Sie Einträge in der Datei über einen FTP Zugriff geändert haben, müssen Sie die ALU320 Aus- und Einschalten, damit die neuen Daten übernommen und die entsprechenden Funktionen aktiviert werden.

6.2 Einstellung in der ALURAS.INI

Projektierung über SPS-Konfigurator "ALU-Parameter –> Grundeinstellungen bearbeiten -> GPRS-VPN Einstellungen", Übertragung per FTP.

Achtung: Die meisten der im Folgenden beschriebenen Parameter dürfen nicht verändert werden. Die für den individuellen Gebrauch relevanten Einstellungen werden über den SPS-Konfigurator in Multiprog abgefragt. Bitte verwenden Sie ausschließlich den SPS-Konfigurator für die Bearbeitung der Datei. Bei nicht sachgemäßen Änderung der Einstellungen kann die OHP GmbH nicht für die einwandfreie Funktionalität garantieren.

```
[AXRASVPN-PARAMS]
PPPDisconnect= FALSE
PPPDisconnect3Att= FALSE
RedialWTimeMS= 5000
FailRedialWTimeMS= 5000
ReConnectWTimeMS= 5000
FailReConnectWTimeMS= 5000
PPPLogoutLoginMode= 1
PPPLogoutLoginWTimeSec= 60
PPPLogoutLoginAt= 12:00:00
PPPCheckWtimeSec= 3600
HwReset3Att= TRUE

[AXRASVPN-PARAMS-T2]
VPNDisable= FALSE
DialRetry= 0
DialRetryWTimeMin= 0
VpnRetry= 0
VpnRetryWTimeMin= 0

[DIAL-LOCATION]
LOCAL= G
DISTANCE= G
INTERNATIONAL= G
AREA=
COUNTRY= 49
DIALTONE= FALSE
DISABLE-WAITING= TRUE
WAITING-CMD=
```



```
[RASMODEM]
Baudrate= 19200
Parity= NO
DataBits= 8
Stopbits= ONE
FlowCtrl= HARD
CancelTime= 120
DialWaitTime= 0
DialCmdOpt= +CGDCONT=1,"IP","internet.t-mobile"

[ISP-PPP-DIAL]
ComPort= 3
CountryCode= 49
AreaCode= 60
LocalPhoneNumber= *99***1#
DialAsLocalCall= FALSE

[ISP-PPP-TCPIP]
IpHeaderCompression= FALSE
SwCompression= FALSE
Slip= FALSE
SpecificIpAddr= FALSE
IpAddr= 0.0.0.0
NameServers= FALSE
ipaddrDns= 0.0.0.0
ipaddrDnsAlt= 0.0.0.0
ipaddrWins= 0.0.0.0
ipaddrWinsAlt= 0.0.0.0

[ISP-PPP-SECURITY]
RequireDataEncryption= FALSE
PAP= FALSE
CHAP= FALSE
MsCHAP= FALSE
MsCHAP2= FALSE
```

```
[ISP-PPP]
UserName=t-mobile
Password=tm
Domain=

[ISP-PPP-PARAMS]
AllowSuspend= 0
AlwaysRequestDNSandWINS= 0
AlwaysSuggestIpAddr= 0
AuthMaxTries= 16
AuthMaxFailures= 3
CryptTypesSupported= 64
MaxConfigure= 16
MaxFailure= 5
MaxTerminate= 2
RestartTimer= 3

[VPN-PPTP]
Hostname= 123.123.123.123
UserName=Beispiel
Password=123456
Domain=

[VPN-PPTP-TCPIP]
IpHeaderCompression= TRUE
SwCompression= FALSE
Slip= FALSE
SpecificIpAddr= FALSE
IpAddr= 0.0.0.0
NameServers= FALSE
ipaddrDns= 0.0.0.0
ipaddrDnsAlt= 0.0.0.0
ipaddrWins= 0.0.0.0
ipaddrWinsAlt= 0.0.0.0

[VPN-PPTP-SECURITY]
RequireDataEncryption= TRUE
```



```
PAP= FALSE
CHAP= FALSE
MsCHAP= FALSE
MsCHAP2= TRUE

[VPN-PPTP-PARAMS]
TcpPortNumber= 1723
UdpPortNumber= 47
MaxWanEndpoints= 5
MaxTransmit= 32
InactivityIdleSeconds= 60
AlwaysEcho= 1
TunnelConfig= 0
TcpDisconnectTimeout= 30
TcpConnectTimeout= 30

[UNIMODEM-INIT]
Init1= AT<cr>
Init2= AT&FE0V1&C1&D2<cr>
Init3=
Init4=
```

6.2.1 Einstellung Parametersatz [AXRASVPN-PARAMS]

PPPDisconnect

Baut die GPRS-Providerverbindung bei jedem fehlgeschlagen Loginversuchen am VPN-Server ab und wieder auf. Wertebereich: TRUE / FALSE.

PPPDisconnect3Att

Baut die GPRS-Providerverbindung nach drei fehlgeschlagen Loginversuchen am VPN-Server ab und wieder auf. Wertebereich: TRUE / FALSE.



Hinweis: Ist PPPDisconnect3Att=FALSE und PPPDisconnect=TRUE, dann wird automatisch PPPDisconnect3Att=TRUE gesetzt!

RedailWTimeMS

Wartezeit in Millisekunden zwischen Einwahlversuchen beim GPRS-Provider (Siehe auch Grafik Grundprinzip Wartezeit). Wertebereich: 5000 - 1342177727

FailRedialWTimeMS

Wartezeit in Millisekunden nach drei fehlgeschlagenen Einwahlversuchen beim GPRS-Provider (Siehe auch Grafik Grundprinzip Wartezeit). Wertebereich: 5000 - 1342177727

ReConnectWTimeMS

Wartezeit in Millisekunden zwischen Loginversuchen am VPN-Server (Siehe auch Grafik Grundprinzip Wartezeit). Wertebereich: 5000 - 1342177727

FailReConnectWTimeMS

Wartezeit in Millisekunden nach drei fehlgeschlagenen Loginversuchen am VPN-Server (Siehe auch Grafik Grundprinzip Wartezeit). Wertebereich: 5000 - 1342177727

PPPLogoutLoginMode

Modus GPRS-Zwangstrennung. Default = 1
0: aus (keine Zwangstrennung)
1: Trennung zu bestimmter Uhrzeit
2: Trennung nach bestimmter Zeitspanne

PPPLogoutLoginWTimeSec

Dauer der GPRS-Zwangstrennung in Sekunden. Default 60 Sekunden

PPPLogoutLoginAt

Uhrzeit bzw. Zeitintervall der GPRS-Zwangstrennung (abhängig von Parameter *PPPLogoutLoginMode*).
Format: hh:mm:ss

PPPCheckWtimeSec

Überwachungsintervall GPRS-Traffic in Sekunden. Falls innerhalb dieser Zeit keine Daten über die GPRS-Verbindung übertragen werden, wird die GPRS-Verbindung getrennt und neu aufgebaut.

HwReset3Att

Legt fest, ob nach dem dritten erfolglosen GPRS-Einwahlversuch ein Hardware-Reset des Modems durchgeführt wird. Wertebereich: TRUE / FALSE. Default=TRUE.

6.2.2 Einstellung Parametersatz [AXRASVPN-PARAMS- T2]

VPNDisable

Legt fest, ob ein VPN Tunnel aufgebaut werden soll. Wertebereich: TRUE / FALSE. Default = FALSE.

DialRetry

Anzahl aufeinander folgender erfolgloser GPRS Einwahlversuche nach dem die Wartezeit "DialRetryWTimeMin" [Minuten] ausgeführt wird. Nach dieser Zeit wird generell der optionale Parameter "HwReset3Att" geprüft. Wertebereich 3…63. Werte kleiner 3 bedeuten aus. Default = 0.

DialRetryWTimeMin

Wartezeit in Minuten nach "DialRetry" fehlgeschlagenen Einwahlversuchen beim GPRS-Provider.
Wertebereich 5 - 1440 Min. Default = 5.

VpnRetry

Anzahl aufeinander folgender erfolgloser Loginversuche am VPN-Server nach dem die Wartezeit "DialRetryWTimeMin" [Minuten] ausgeführt wird. Nach dieser Zeit wird generell der optionale Parameter "HwReset3Att" geprüft. Wertebereich 3…63. Werte kleiner 3 bedeuten aus. Default = 0.

VpnRetryWTimeMin

Wartezeit in Minuten nach "VpnRetry" fehlgeschlagenen Loginversuchen am VPN-Server.
Wertebereich 10 - 1440 Min. Default = 10.



6.2.3 Einstellung Parametersatz [DIAL-LOCATION]

LOCAL

Amtskennziffer für Ortsverbindung. Wertebereich G, E, F. Default = G.

DISTANCE

Amtskennziffer für Fernverbindung. Wertebereich G, E, F. Default = G.

INTERNATIONAL

Amtskennziffer für Auslandsverbindung. Wertebereich G, E, F. Default = G.

AREA

Vorwahl. Wertebereich alphanumerische Zeichen.

COUNTRY

Ländervorwahl. Wertebereich alphanumerische Zeichen. Default =49.

DIALTONE

Wählverfahren. Default = TRUE.

TRUE: Frequenzverfahren (MFV),
FALSE: Impulswahlverfahren

DISABLE-WAITING

Sperre nachfolgendes WAITING-CMD. Wertebereich TRUE / FALSE. Default=TRUE.

WAITING-CMD

LASS Code (Local Area Signaling Services) oder/und benutzerdefinierte Calling Feature Control Codes:
z.B. *70 ... Call Waiting deaktivieren.
Wertebereich: Frei oder ASCII-Zeichen. Default = frei.

6.2.4 Einstellung Parametersatz [RASMODEM]

Baudrate

Geschwindigkeit RS232-Schnittstelle zu GPRS-Modem. Einstellung = 19200.

Parity

Parität RS232-Schnittstelle zu GPRS-Modem: Einstellung=NO (kein Paritybit)

DataBits

Anzahl Datenbits RS232-Schnittstelle zu GPRS-Modem. Einstellung = 8.

Stopbits

Anzahl Stopbits RS232-Schnittstelle zu GPRS-Modem. Einstellung ONE (1 Stoppbit)

FlowCtrl

Datenflusskontrolle RS232-Schnittstelle zu GPRS-Modem. Einstellung = HARD (Hardware)

CancelTime

Zeit in Sekunden, nach der die GPRS-Einwahl abgebrochen wird, falls keine Verbindung zustande kommt.

Einstellung = 120.

DialWaitTime

Wartezeit auf Freizeichen in Sekunden. Einstellung = 0 (Aus).

DialCmdOpt

PDP Kontext mit APN (Access Point Name) des GPRS-Providers.

Hier muss nur der APN des jeweiligen Providers eingetragen werden (letztes Feld). Dieser wird Ihnen mitgeteilt, wenn Sie z.B. einen speziellen Vertrag abgeschlossen haben, bei dem Ihnen der Provider einen VPN-Tunnel einrichtet. Standard-APNs finden Sie auf den Webseiten Ihres Anbieters.

Default = +CGDCONT=1,"IP","internet.t-mobile".

6.2.5 Einstellung Parametersatz [ISP-PPP-DIAL]

ComPort

Nummer des seriellen Anschlusses des GPRS-Modems. Bei der ALU320 mit integriertem Modem ist dies immer COM3.

CountryCode

Ländervorwahl. Wertebereich Keine Angabe oder alphanumerische Zeichen. Default = 49 (Deutschland).

AreaCode

Vorwahl. Wertebereich Keine Angabe oder alphanumerische Zeichen.

LocalPhoneNumber

GPRS-Einwahlnummer (abhängig von GPRS-Provider). Einstellung = *99***1#.

DialAsLocalCall

Wertebereich TRUE/FALSE . Default = FLASE.

6.2.6 Einstellung Parametersatz [ISP-PPP-TCPIP]

IpHeaderCompression

Aktiviert IP Header Compression auf GPRS-Verbindungsebene. Wertebereich TRUE/FALSE. Default= FLASE.

SwCompression

Aktiviert Software-Compression auf GPRS-Verbindungsebene. Wertebereich TRUE/FALSE. Default= FLASE.

Slip

Aktiviert *Serial Line Internet Protocol* auf GPRS-Verbindungsebene.
Wertebereich TRUE/FALSE. Default= FLASE.

SpecificIpAddr

Legt fest, ob auf GPRS-Verbindungsebene eine spezielle IP-Adresse verwendet werden soll.
Wertebereich TRUE/FALSE. Default= FLASE.

IpAddr

Legt die IP-Adresse für die GPRS-Verbindung fest. Falls *SpecificIpAddr=FALSE* wird dieser Parameter ignoriert

NameServers

Legt fest, ob auf GPRS-Verbindungsebene spezielle IP-Adressen für die Namensauflösung verwendet werden sollen. Wertebereich TRUE/FALSE. Default= FLASE.



ipaddrDns

IP-Adresse des DNS-Servers (wird ignoriert falls *NameServers=FALSE*)

ipaddrDnsAlt

Alternative IP-Adresse des DNS-Servers (wird ignoriert falls *NameServers=FALSE*)

ipaddrWins

IP-Adresse des WINS-Servers (wird ignoriert falls *NameServers=FALSE*)

ipaddrWinsAlt

Alternative IP-Adresse des WINS-Servers (wird ignoriert falls *NameServers=FALSE*)

6.2.7 Einstellung Parametersatz [ISP-PPP-SECURITY]

RequireDataEncryption

Legt fest, ob GPRS-Provider zwingend Datenverschlüsselung anbieten muss. Wertebereich TRUE/FALSE. Default= FLASE.

PAP

Legt fest, ob auf GRPS-Verbindungsebene das Passwort Authentication Protocol (PAP) akzeptiert wird. Wertebereich TRUE/FALSE. Default= FLASE.

CHAP

Legt fest, ob auf GRPS-Verbindungsebene das Challenge Handshake Authentication Protocol (CHAP) akzeptiert wird. Wertebereich TRUE/FALSE. Default= FLASE.

MsCHAP

Legt fest, ob auf GRPS-Verbindungsebene das Microsoft Challenge Handshake Authentication Protocol (MSCHAP) akzeptiert wird. Wertebereich TRUE/FALSE. Default= FLASE.

MsCHAP2

Legt fest, ob auf GRPS-Verbindungsebene das Microsoft Challenge Handshake Authentication Protocol Version 2.0 (MSCHAPv2) akzeptiert wird. Wertebereich TRUE/FALSE. Default= FLASE.

6.2.8 Einstellung Parametersatz [ISP-PPP]

UserName

GPRS-Benutzername (abhängig von GPRS-Provider). Muss nicht bei allen Providern angegeben werden.
Wertebereich: Frei oder maximal 16 ASCII-Zeichen. Default=t-mobile

Password

GPRS-Kennwort (abhängig von GPRS-Provider). Muss nicht bei allen Providern angegeben werden.
Wertebereich: Frei oder maximal 16 ASCII-Zeichen. Default=tm.

Domain

GPRS-Anmeldedomäne. Muss nicht bei allen Providern angegeben werden.
Wertebereich: Frei oder maximal 16 ASCII-Zeichen. Default=<frei>.

6.2.9 Einstellung Parametersatz [ISP-PPP-PARAMS]

AllowSuspend

Bestimmt das Unterbrechungsverhalten des Treibers. Um eine automatische Unterbrechung einer aktiven PPP-Verbindung durch den Treiber zu erlauben, muss der Eintrag auf 1 gesetzt werden.
Wertebereich TRUE/FALSE. Einstellung=FALSE.

AlwaysRequestDNSandWINS

Legt fest, ob der Client DNS und WINS Adressen vom Server abfragt, auch wenn statische Adressen eingetragen sind. Wertebereich TRUE/FALSE. Einstellung=FALSE.

AlwaysSuggestIpAddr

Bei Einstellung TRUE schlägt der Client immer eine IP-Adresse vor, auch wenn keine statische IP-Adresse vorgegeben ist. Bei Einstellung FALSE fordert der Client die IP-Adresse vom Server an.
Wertebereich TRUE/FALSE. Einstellung=FALSE.

AuthMaxTries

Maximale Anzahl von Versuchen des Windows CE RAS Servers den Client zu authentifizieren.
Wertebereich 1…16. Einstellung =16.

AuthMaxFailures

Gibt die maximale Anzahl von erfolglosen Authentifizierungen an, bevor der Windows CE RAS Server die Verbindung unterbricht. Wertebereich 1…3. Einstellung =3.

CryptTypesSupported

Unterstützte Verschlüsselungsverfahren. Einstellung=64.

0..Verschlüsselung deaktiviert

32..aktiviert 40-Bit MPPE-Verschlüsselung

64..aktiviert 128-Bit MPPE-Verschlüsselung

96..aktiviert 40-Bit und 128-Bit MPPE-Verschlüsselung

MaxConfigure

Legt die maximale Anzahl von gesendeten Configure-Request (CR) Paketen fest, die nicht durch ein Configure-ACK, Configure-No Acknowledgement (NAK) oder Configure-Reject bestätigt wurden, nach der davon ausgegangen wird, das die Gegenstelle nicht antwortet. Wertebereich 1…16. Einstellung =16.

MaxFailure

Legt die maximale Anzahl von Configure-NAK-Paketen fest, die nicht durch ein Configure-ACK bestätigt wurden, nach der davon ausgegangen wird, dass die Konfiguration nicht übereinstimmt.
Wertebereich 1…5. Einstellung =5.

MaxTerminate

Legt die maximale Anzahl der Terminate-Request Pakete fest, die nicht durch ein Terminate-ACK bestätigt wurden, nach der davon ausgegangen wird, das die Gegenstelle nicht antwortet. , nach der davon ausgegangen wird, das die Gegenstelle nicht antwortet. Wertebereich 1…16. Einstellung =16.
Wertebereich 1…5. Einstellung =2.

RestartTimer

Gibt die Wartezeit in Sekunden an, bevor eine fehlgeschlagener Configure-Request oder Terminate-Request wiederholt wird.



6.2.10 Einstellung Parametersatz [VPN-PPTP]

Hostname

IP-Adresse des VPN-Servers. Werteberich: IP-Adresse

UserName

VPN-Benutzername. Wertebereich max. 16 ASCII-Zeichen

Password

VPN-Kennwort: Wertebereich: ASCII-Zeichen

Domain

VPN-Anmeldedomände. Wertebereich: ASCII-Zeichen

6.2.11 Einstellung Parametersatz [VPN-PPTP-TCPIP]

IpHeaderCompression

Aktiviert IP Header Compression auf VPN-Verbindungsebene. Wertebereich TRUE/FALSE. Default= TRUE.

SwCompression

Aktiviert Software-Compression auf VPN-Verbindungsebene. Wertebereich TRUE/FALSE. Default= FLASE.

Slip

Aktiviert *Serial Line Internet Protocol* auf VPN-Verbindungsebene.
Wertebereich TRUE/FALSE. Default= FLASE.

SpecificIpAddr

Legt fest, ob auf VPN-Verbindungsebene eine spezielle IP-Adresse verwendet werden soll.
Wertebereich TRUE/FALSE. Default= FLASE.

IpAddr

Legt die IP-Adresse für die VPN-Verbindung fest. Falls *SpecificIpAddr=FALSE* wird dieser Parameter ignoriert

NameServers

Legt fest, ob auf VPN-Verbindungsebene spezielle IP-Adressen für die Namensauflösung verwendet werden sollen. Wertebereich TRUE/FALSE. Default= FLASE.

ipaddrDns

IP-Adresse des DNS-Servers auf VPN-Verbindungsebene (wird ignoriert falls *NameServers=FALSE*)

ipaddrDnsAlt

Alternative IP-Adresse des DNS-Servers auf VPN-Verbindungsebene (wird ignoriert falls *NameServers=FALSE*)

ipaddrWins

IP-Adresse des WINS-Servers auf VPN-Verbindungsebene (wird ignoriert falls *NameServers=FALSE*)

ipaddrWinsAlt

Alternative IP-Adresse des WINS-Servers auf VPN-Verbindungsebene (wird ignoriert falls *NameServers=FALSE*)

6.2.12 Einstellung Parametersatz [VPN-PPTP-SECURITY]

RequireDataEncryption

Legt fest, ob der VPN-Server zwingend Datenverschlüsselung anbieten muss. Wertebereich TRUE/FALSE. Default= TRUE.

PAP

Legt fest, ob auf VPN-Verbindungsebene das Passwort Authentication Protocol (PAP) akzeptiert wird. Wertebereich TRUE/FALSE. Default= FLASE.

CHAP

Legt fest, ob auf VPN-Verbindungsebene das Challenge Handshake Authentication Protocol (CHAP) akzeptiert wird. Wertebereich TRUE/FALSE. Default= FLASE.

MsCHAP

Legt fest, ob auf VPN-Verbindungsebene das Microsoft Challenge Handshake Authentication Protocol (MSCHAP) akzeptiert wird. Wertebereich TRUE/FALSE. Default= FLASE.

MsCHAP2

Legt fest, ob auf VPN-Verbindungsebene das Microsoft Challenge Handshake Authentication Protocol Version 2.0 (MSCHAPv2) akzeptiert wird. Wertebereich TRUE/FALSE. Default= TRUE.

6.2.13 Einstellung Parametersatz [VPN-PPPT-PARAMS]

TcpPortNumber

TCP-Portnummer des PPTP-Zugangs des VPN-Servers Wertebereich 1…65535. Einstellung 1723

UdpPortNumber

UDP-Portnummer des VPN-Servers. Wertebereich 1…65535. Einstellung 47

MaxWanEndpoints

Maximale Anzahl der PPTP-Sessions. Einstellung 5.

MaxTransmit

Maximale Anzahl ausstehender PPTP-Sendepakete. Einstellung 32.

InactivityIdleSeconds

Timeout-Intervall in Sekunden, nach denen ein CCP-Paket (Control channel packet) an den VPN-Server gesendet wird, falls in dieser Zeit keine Pakete durch den Tunnel übertragen wurden. Einstellung=60.

AlwaysEcho

Wertebereich 0/1. Einstellung = 1 (schaltet den Echo-Modus ein).

TunnelConfig

Legt die Routing-Konfiguration für das Tunneln von Clients fest. Der Eintrag ist auf 0 zu stellen.

TcpDisconnectTimeout

Legt die maximale Wartezeit für die Bestätigung eines Verbindungsabbaus auf TCP-Ebene fest. Einstellung=30.

TcpConnectTimeout

Legt die Time-Out-Zeit für eine TCP Verbindung fest. Einstellung=30.



6.2.14 Einstellung Parametersatz [UNIMODEM-INIT]

Init1, Init2, Init3, Init4

Die komplette Konfiguration des internen Modems erfolgt über AT-Kommandos, die vom Betriebssystem in Form der Initialisierungsstrings an das Modem gesendet werden, es können mehrere Kommandos nacheinander eingegeben werden. Jede Zeichenkette beginnt mit AT und endet mit <cr>.

AT&FE0V1&C1&D2<cr>

&F Werksvoreinstellung laden

E0 Echo ausschalten

V1 Meldung in Textform

&C1 DCD zeigt an, dass eine Verbindung aufgebaut und synchronisiert ist.

&D2 Bei fallender Flanke am DTR wird die Verbindung unterbrochen.

6.2.15 Konfigurationsbeispiel

Parametersatz [AXRASVPN_PARAMS]

RedialWTimeMS= 15000 // Alle 15 Sekunden neuen ISP-Einwahlversuch starten

FailRedialWTimeMS= 105000 // Aber nach 3 erfolglosen ISP-Einwahlversuchen, 105s warten

ReConnectWTimeMS= 120000 // Alle 120 Sekunden neuen VPN-Einwahlversuch starten

FailReConnectWTimeMS= 1800000 // Aber nach 3 erfolglosen VPN-Einwahlversuchen ISP-Verbindung beenden und 30 Min. warten

Parametersatz [AXRASVPN_PARAMS-T2]

.

DialRetry=9 // Nach insgesamt 9 erfolglosen ISP-Einwahlversuchen…

DialRetryWTimeMin=5 // …erst mal 5 Minuten warten

VpnRetry=9 // Nach insgesamt 9 erfolglosen VPN – Einwahlversuchen…

VpnRetryWTimeMin=480 // …erst mal 480 Min (8Stunden) warten

Wartezeit für Verbindungsaufbau: Modem an GPRS – Provider

Erfolglose Einwahl-versuche	1	2	3	4	5	6	7	8	9	10
Zeiten		ts	ts	tb	ts	ts	tb	ts	ts	tdr

ts = RedialWTimeMS tb = FailRedialWTimeMS tdr = DialRetryWTimeMS

Wartezeit für Verbindungsaufbau: VPN Client (ALU) – VPN Server (Router)

Erfolglose Einwahl-versuche	1	2	3	4	5	6	7	8	9	10
Zeiten		tnc	tnc	trc	tnc	tnc	trc	tnc	tnc	tvr

tnc = ReConnectWTimeMS trc = FailReConnectWTimeMS tdr = VpnRetryWTimeMin



6.3 Freigabe der Service-Routinen und Interfaces

Die einzelnen Service-Routinen und Schnittstellen werden über die Datei PCOS_A320.INI frei gegeben. Die Datei befindet sich im Verzeichnis PCOS_OHP auf der CF-Karte.



Hinweis: Ab Firmware-Version 1.70 werden geänderte Dateinamen verwendet (PCOSA320.INI)

Bei Auslieferung der CF-Karte sind alle Einträge inaktiv. Durch Entfernen des Doppelschrägstrichs // wird ein Eintrag freigegeben.

Achtung: Schalten Sie in Ihrem Projekt nur die Service-Routinen und Schnittstellen (Interface) frei, die Sie tatsächlich benötigen. Anhand der Einstellungen in dieser Datei werden beim Hochlauf der ALU 320 die Treiberdateien (DLLs) in den Speicher geladen. Jeder freigegebene Treiber belegt Speicherplatz, auch dann, wenn er nicht verwendete wird.

<table>
<tr><td colspan="2">#==
#
ALU320 PCOS Einstellungen
ALU320 PCOS settings
#
#==</td></tr>
<tr><td>[PCOS-SYSTEM]
SharedMemoryAddr=1000
SharedMemorySize=600
TimeSyncDelay=0</td><td>#Speicherbereich für Shared-Memory
#Startadresse ProConOs %MB3.1000
#Anzahl Bytes in Summe   bis %MB3.1599
#Gangreserve in Stunden (1…72), 0 = aus</td></tr>
<tr><td colspan="2">#==
#
MODBUS Server Einstellungen: Speicher, Schnittstellen
max. 4 Schnittstellen möglich
#
MMODBUS server settings: memory, interface
max. 4 interface are possible
#
#==</td></tr>
<tr><td>[MODBUS-SERVER]

//SharedMemoryOffset=200
//SharedMemorySize=308

//Interface1= SSERIAL
//Interface2= STCP_IP</td><td>#Speicherbereich für Modbus

#Startadresse
#Anzahl Bytes in Summe

#Slave Serial
#Modbus TCP-Server</td></tr>
<tr><td colspan="2">
#==
#
Service-Routinen für IEC 60870-5-101 oder IEC 60870-5-104
#
#==</td></tr>
<tr><td>[SERVICE]
//Service1= SVC104SA
//Service2= SVC101SA</td><td>#Protokoll IEC 60870-5-104 (TCP/IP)
#Protokoll IEC 60870-5-101 (seriell)</td></tr>
</table>

Beispiel:

Mit der auf der CF-Karte enthaltenen Beispielprojektierung kann durch Freigabe von Interface1 und Service2 z.B. eine Datenübertragung nach IEC 60870-5-101 über die COM1 und eine Ankopplung an einen Modbus-Master über die COM2 der ALU 320 realisiert werden.

6.3.1 Definition von Shared Memory

Die Startadresse des Shared Memory kann bei 0 (%MB3.000) beginnen. Bitte achten Sie darauf als Startadresse immer gerade Adressen zu verwenden (0, 2, 4, 200, 1000 etc.)

Die Länge kann bis maximal 16384 vorgegeben werden. Geben Sie auch bei der Länge eine gerade Zahl ein.

Im Beispiel steht dem Anwenderprogramm (AWP) ein Shared Memory Bereich von %MB3.1000 bis %MB3.1599 zur Verfügung.

Dieser kann z.B dafür benutzt werden, um überlappende Variablen oder Strukturen zu definieren.

Beispiel:

Variable A	BYTE	%MB3.1200
Variable A1	BOOL	%MX3.1200.0
.		
.		
Variable A8	BOOL	%MX3.1200.7

Der gesamte Shared Memory Bereich oder Teile davon können für eine Modbus-Ankopplung verwendet werden.

6.3.2 Modbus RS232

Werden die Einträge für den Modbus durch Löschen der // freigegeben, kann im Beispiel der Bereich %MB3.1200 bis %MB3.1507 über die Modbus RS232-Schnittstelle beschrieben und gelesen werden.

[MODBUS-SERVER]

SharedMemoryOffset=200
SharedMemorySize=308

Interface1= **SSERIAL**

Sie können auch den kompletten Shared Memory Bereich für die Modbus-Übertragung zur Verfügung stellen. Im Beispiel wäre das mit folgenden Einstellungen möglich:

[MODBUS-SERVER]

SharedMemoryOffset=0
SharedMemorySize=600

Interface1= **SSERIAL**

Über die Datei MODBUS_SERVER.INI wird dieser Bereich auf die einzelne Coils und Register zugewiesen.

📖 Weiterführende Informationen siehe Kapitel 6.5.



6.3.3 Modbus TCP-Server

Werden die Einträge für den Modbus TCP-Server durch Löschen der // beim Interface2 freigegeben, kann im Beispiel der Bereich %MB3.1200 bis %MB3.1507 über Modbus TCP beschrieben und gelesen werden.

[MODBUS-SERVER]

SharedMemoryOffset=200
SharedMemorySize=308

Interface2= **STCP_IP**

Sie können auch den kompletten Shared Memory Bereich für die Modbus-Übertragung zur Verfügung stellen. Im Beispeil wäre das mit folgenden Einstellungen möglich:

[MODBUS-SERVER]

SharedMemoryOffset=0
SharedMemorySize=600

Interface2= **STCP_IP**

Über die Datei MODBUS_SERVER.INI wird dieser Bereich auf die einzelne Coils und Register zugewiesen.

 Weiterführende Informationen siehe Kapitel 6.5.

6.3.4 Service-Routinen

Soll eine Kommunikation über IEC 60870-5-101 und/oder IEC 60870-5-104 aufgebaut werden, sind die entsprechenden Service-Routinen freizugeben. Es ist möglich mehrere Routinen gleichzeitig zu starten.

Die Einstellung

Service1= SVC104SA (zugehörige Parameter-Dateien SVC_COM1.INI und SVC_RTU1.INI)

Service2= SVC101SA (zugehörige Parameter-Dateien SVC_COM2.INI und SVC_RTU2.INI)

ermöglicht die gleichzeitige Übertragung von Daten über TCP/IP und eine serielle Ankopplung.

Die Einstellung

Service1= SVC101SA (zugehörige Parameter-Dateien SVC_COM1.INI und SVC_RTU1.INI)

Service2= SVC101SA (zugehörige Parameter-Dateien SVC_COM2.INI und SVC_RTU2.INI)

ermöglicht die gleichzeitige Übertragung der Daten über die beiden seriellen Schnittstellen der ALU320 mit IEC 60870-5-101 Protokoll.

Anmerkung: Bei Auslieferung der CF-Karte ist die Datei SVC_COM1.INI für das Protokoll IEC 60870-5-104 vorbereitet. Wenn wie oben beschrieben beide Service-Routinen mit dem seriellen Protokoll arbeiten sollen, kopieren Sie den Inhalt der Datei SVC_COM2.INI in die SVC_COM1.INI und ändern Sie die Einstellungen entsprechend Ihrer Anwendung ab. Achten sie insbesondere darauf, dass die Einstellung der Ports in den beiden Dateien unterschiedlich ist.

 **Hinweis:** Ab Firmware-Version 1.70 werden geänderte Dateinamen verwendet (SVCCOMn.INI)

6.4 IEC 60870-5-101 oder -104 Kommunikation

6.4.1 Allgemeines

Für die einzelnen Übertragungsprotokolle können auf der ALU320 bis zu 8 Service-Routinen gestartet werden. Jeder Service-Routine werden Parameterdateien für die Kommunikation (SVC_COM?.INI) und die Definition der PV-Verarbeitung (SVC_RTU?.INI) zugeordnet.

Es sind die Protokolle IEC 60870-5-101 und IEC 60870-5-104 mit Slave-Funktionalität verfügbar.

Die Parameterdateien SVC_COM?.INI und SVC_RTU?.INI befinden sich im Verzeichnis PCOS-OHP auf der CF-Karte.



Hinweis: Ab Firmware-Version 1.70 werden geänderte Dateinamen verwendet (SVCCOMn.INI und SVCRTUn.INI)

Anmerkung: Die einzelnen Parameter der auf der CF-Card enthaltenen Dateien können von dieser Darstellung abweichende Zuweisungen haben.

Ändern Sie die Einträge entsprechend Ihrer Aufgabenstellung und speichern Sie die Dateien anschließend wieder auf der CF-Card.



Hinweis: Wird einem Eintrag ein Doppel-Schrägstrich "//" vorangestellt, wird die entsprechende Zeile nicht interpretiert. Dies kommt einem Löschen der Zeile gleich und der entsprechende Parameter ist nicht aktiviert. Nicht benötigte Parameter können gelöscht oder deaktiviert werden.
Die Defaulteinstellung für gelöschte oder deaktivierte Parameter ist 0, FALSE oder OFF.

Der Eintrag TRUE entspricht ON, der Eintrag FALSE entspricht OFF. Es können jeweils beide Varianten der Bezeichnungen verwendet werden.



6.4.2 Parameterdatei für die Kommunikationsprojektierung IEC 60870-5-104 (SVC_COMn)



Hinweis: Ab Firmware-Version 1.70 werden geänderte Dateinamen verwendet (SVCCOMn.INI)

[IEC-5-104]	
PORT= 2404	Portnummer **Darf nicht geändert werden!**
KVAL= 12	Maximale Differenz zwischen Anzahl Empfangsfolgen und Sendefolgen
WVAL= 8	Späteste Quittierung nach Empfang von w APDU im I-Format
T1= 15	Zeitüberwachung für gesendete APDU oder Test-APDU
T2= 10	Zeitüberwachung für Quittierungen, falls keine Daten übertragen werden
T3= 40	Zeitüberwachung für gesendete Testframes
IP_ADDR=192.168.0.1	IP-Adresse der Gegenstelle

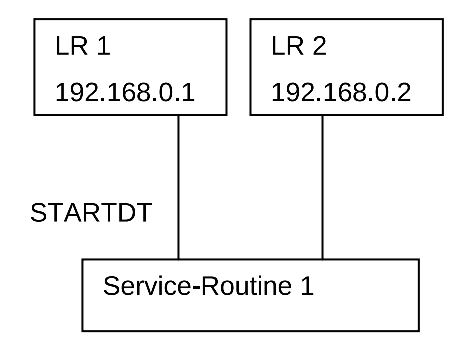
Über eine Service-Routine können maximal zwei IEC 60870-5-104 Verbindungen aufgebaut werden. Jedoch darf nur bei **einer** Verbindung das STARTDT aktiviert werden. Bei der 2. Verbindung werden nur die Testframes (TESTFR) bestätigt.

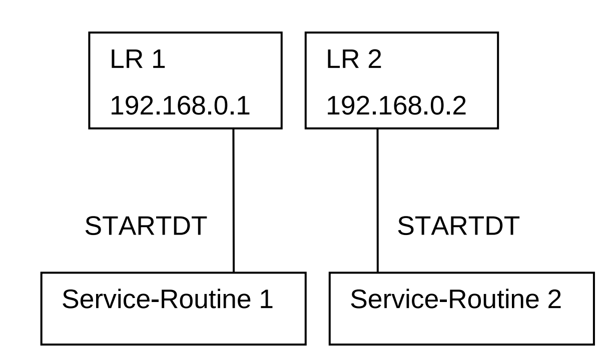
Sie können die IP-Adresse des Kommunikationspartners eintragen (max. 2). Damit wird sichergestellt, dass die ALU320 eine TCP/IP-Verbindung nur zu dieser Adresse zulässt. Verbindungsversuche von anderen Adressen werden nicht bestätigt.

Wird keine Adresse eingetragen, kann über jede beliebige IP-Adresse im gleichen Subnetz eine Verbindung e-tabliert werden.



Hinweis: Werden für eine redundante Datenübertragung zwei Verbindungen, mit aktiviertem STARTDT benötigt, müssen zwei Service-Routinen projektiert werden.





Daten werden nur mit LR1 ausgetauscht.

Bei LR2 werden nur TESTFR beantwortet.

Ein Wechsel kann nur durch STOPDT bei LR1 und anschließendem STARTDT bei LR2 ver-anlasst werden

Daten werden mit beiden LR ausgetauscht

6.4.3 Parameterdatei für die Kommunikationsprojektierung IEC 60870-5-101 (SVC_COMn.INI)

Öffnen Sie die Datei SVC_COM.INI die Sie auf Ihrem Rechner gespeichert haben mit dem Editor.

 **Hinweis:** Ab Firmware-Version 1.70 werden geänderte Dateinamen verwendet (SVCCOMn.INI)

Sie sehen jetzt folgende Einträge:

[IEC-5-101]		
AWD= FALSE	TRUE = AWD-Betrieb, FALSE = Standleitungsbetrieb	
PORT= 1	Port = 1 oder 2 für externe, 3 für interne COM-Schnittstelle	
BAUDRATE= 9600	Übertragungsrate zum Leitrechner	
PARITY= EVEN	Parität NO / EVEN	(bei Standleitung immer EVEN)
DATA=8	Anzahl Datenbits	**Darf nicht geändert werden!**
STOPBIT=1	Stoppbit	**Darf nicht geändert werden!**
TVS2=25	Vorlaufzeit S2 [ms]	(nur relevant bei Standleitung)
TNS2=7	Nachlaufzeit S2 [ms]	(nur relevant bei Standleitung)
TVM2=40	Vorlaufüberwachung M2 [ms]	(nur relevant bei Standleitung)
TNM2=150	Nachlaufüberwachung M2 [ms]	(nur relevant bei Standleitung)
PAUSE=27	Pausenzeit [ms]	
BUS_TOUT= 2	Busüberwachung (Slave) in n*[100ms]	
POLL_TOUT= 60	Überwachung Pollzyklus (Slave) in n*[100ms]	
S2SIGNAL= OFF	S2-Signal (RTS) ein- oder ausschalten	
M2SIGNAL= OFF	M5/M2-Überwachung (DCD/CTS) ein- oder ausschalten	
CONF_SC= OFF	ON= Empfangsbestätigung mit Einzelzeichen E5, OFF = Kurztelegramm	
AVS= 3	Adresse der Verbindungsschicht	
OKTETT_AVS= 2	Anzahl Oktett der AVS (1 bis 2)	
[AWD101-MODE]	!! AT Befehle ASCII, max. 8 Zeichen !!	
AT_INIT= ATZ0	Abgespeichertes Profil in Modem laden	
AT_HANGUP= ATH	AT-Kommando für Auflegen	
AT_DIAL= ATD	AT-Kommando für Wählen	
AT_PICK_UP= ATA	AT-Kommando für Abheben	
# Modem RESPONSE/Rückmeldung Dezimal		
//RSP_OK= 0	Hier sind die bei den Modems üblichen Werte für die Rückmeldungen angegeben. Sollte das von Ihnen verwendete Modem davon abweichende Werte verwenden, löschen Sie vor dem entsprechenden Eintrag den Doppel-Slash "//" und geben Sie den verwendeten Wert an.	
//RSP_CONNECT= 1		
//RSP_RING= 2		
//RSP_NOCARRIER= 3		
//RSP_ERROR= 4		
//RSP_NODIAL= 5		
//RSP_BUSY= 6		



//RSP_NOANSWER=7
RSP_TOUT= 2
	Maximale Wartezeit auf die Bestätigung der AT-Kommandos (1 bis 15 Sek.)

# Wahlparameter / Verbindungsaufbau	
DIAL_NUMBER= ????????? DIAL_RETRY= 3 DIAL_TM_REDIAL= 30 DIAL_TM_RIDLE= 1 DIAL_TM_CON= 60 DIAL_TM_PW= 2 DIAL_PRI_PW= ??????? DIAL_SEK_PW= ??????? MODEM_LED=FALSE	Telefonnummer der Zentrale Wahlwiederholung (1 bis 7) Zeit[Sek]: Pause bei Wahlwiederholung (1 bis 255) Zeit[Min]: Sperrzeit nach Wahlwiederholung (1 bis 255) Zeit[Sek]:Wartezeit auf Verbindung "Connect" (1 bis 255) Zeit[Sek]:Wartezeit bis Passwort nach Verbindungsaufbau gesendet wird Passwort Primärstation (Leitrechner) maximal 16 Zeichen Passwort Sekundärstation (Unterstation) maximal 16 Zeichen Mit TRUE wird die Anzeige für die Interne COM3 eingeschaltet

Anmerkung: Die einzelnen Parameter können von dieser Darstellung abweichende Zuweisungen haben.

6.4.3.1 Erläuterung zu den einzelnen Parametern im Standleitungs-Betrieb

PORT

Hier geben Sie die COM-Schnittstelle (1 oder 2) an, die Sie für die Kommunikation mit IEC 60870-5-101 verwenden wollen. Wenn Sie eine ALU mit RS485-Schnittstelle oder integriertem Wählmodem benutzen, werden diese über die COM3 angesprochen.

BAUDRATE

Die Baudrate kann zwischen 600 und 9600 Baud eingestellt werden.

PARITY, DATA, STOPBIT

Bei Standleitungsbetrieb oder direkter serieller Kopplung ist die PARITY auf EVEN zu stellen. Die Anzahl Datenbit muss immer 8 sein, die Anzahl Stoppbit immer 1.

TVS2, TNS2, TVM2, TNM2, PAUSE, S2SIGNAL, M2SIGNAL

Über diese Parameter werden die Vor- und Nachlaufzeit für S2 und die Überwachungszeiten für das M2-Signal, sowie die Pausenzeit zwischen den Telegrammen eingestellt. Diese Angaben sind vor allem bei Verwendung von Standleitungsmodems (UEM 202 oder 201) erforderlich. Über die Parameter S2SIGNAL, M2SIGNAL kann das Setzen bzw. Überwachen der jeweiligen Signale abgeschaltet werden.

POLL_TOUT

Innerhalb dieser Überwachungszeit muss die Station angepollt werden, ansonsten wird die Verbindung als gestört gemeldet und kann nur durch ein "Initialisieren der Verbindungsschicht" aktiviert werden.

BUS_TOUT

Die Überwachung **BUS_TOUT** gibt die Überwachungszeit beim Linienbetrieb an, innerhalb der die eigene oder eine andere Station an der Linie (BUS) gepollt werden muss. Ist die Station in diesen Timout gelaufen, erwartet sie ein Initialisieren der Verbindungsschicht.
Bei Punkt-zu-Punkt oder AWD-Betrieb darf der **BUS_TOUT** nicht kleiner als der **POLL_TOUT** eingestellt werden.

CONF_SC

Über diesen Parameter geben Sie vor, wie die Unterstation antworten soll, wenn keine Datentelegramme zur Übertragung anstehen. ON= Empfangsbestätigung mit Einzelzeichen E5, OFF = Kurztelegramm (Telegram fester Länge).

AVS, OKTETT_AVS

AVS ist die Adresse der Verbindungsschicht. Mit dieser wird die Station gepollt. Die Adresse kann 1 oder 2 Oktett (Byte) lang sein. Diese Einstellung muss mit allen Teilnehmern im System über die Kompatibilitätsliste festgelegt werden.

6.4.3.2 Erläuterung zu den einzelnen Parametern im AWD-Betrieb

PARITY, DATA, STOPBIT

Bei den meisten Wähl-Modems oder ISDN-TA ist die Übertragung von 8 Datenbit nur ohne Paritätsbit möglich. Setzen Sie in diesem Fall den Parameter PARITY= NO. Die Einstellung muss mit der Projektierung auf der Gegenseite übereinstimmen.

AT_INIT, AT_INI2, AT_INI3

Es können bis zu 3 Initialisierungssequenzen mit jeweils maximal 28 Zeichen für Wählmodems eingetragen werden. Dies ist insbesondere für den Betrieb mit internen Modems wichtig, da diese nicht durch externe Tools initialisiert werden können.

Einstellungen für interne Modems:

AT_INIT=ATE0X0

AT_INIT2=ATS0=0V0

Anmerkung: AT_INIT2 und AT_INIT3 stehen ab ALU-Firmwareversion 1.90 zur Verfügung.

RSP_TOUT ist die Überwachungszeit, innerhalb der die Modem-Rückmeldungen (CONNECT, OK etc.) eintreffen müssen. Die vorgegebene Standardeinstellung ist für die meisten Modemtypen passend und muss nicht verändert werden.

Kommt eine von der Unterstation eingeleitete Verbindung nicht innerhalb der mit Parameter **DIAL_TM_CON** angegebenen Zeit zustande, wird der Anwahlversuch abgebrochen.
Nach Ablauf der Wartezeit **DIAL_TM_REDIAL** wird erneut versucht die Verbindung aufzubauen. Dies wird solange wiederholt, bis die Verbindung zustande kommt oder die eingestellte Anzahl von Wahlwiederholungen **DIAL_RETRY** erreicht ist.

Ist die vorgegebene Anzahl von Anwahlversuchen erreicht ohne dass eine Verbindung zustande kam, wird ein Timer mit der in **DIAL_TM_RIDLE** angegebenen Zeit gestartet. Nach Ablauf dieser Zeit wird die oben beschriebene Prozedur erneut durchlaufen.

Das Passwort für die Zentrale (**DIAL_PRI_PW)** und die Unterstation (**DIAL_SEK_PW**) kann maximal 16 Zeichen lang sein. Wird das Passwort mit weniger Zeichen angegeben, wird es automatisch mit Leerzeichen aufgefüllt. Groß/Kleinschreibung wird geprüft. Die angegebenen Passwörter müssen mit der Projektierung im Leitsystem übereinstimmen.

Nachdem ein Verbindungsaufbau erfolgreich eingeleitet wurde (Connect vom Modem) sendet der Leitrechner bzw. die Unterstation sein/ihr Passwort. Das Passwort wird immer von der Seite gesendet, die den Verbindungsaufbau eingeleitet hat und von der Gegenseite geprüft. Ist die Prüfung erfolgreich, wird der Pollbetrieb aufgenommen. Wird bei der Passwortprüfung ein Fehler erkannt, unterbricht die prüfende Seite die Verbindung mit ATH.



MODEM_LED

Dieser Schalter kann auf TRUE gesetzt werden, wenn eine ALU mit integriertem Modem für den Wählbetrieb verwendet wird. Durch die Aktivierung hat die Anzeige der 3. LED folgende Bedeutung:

LED	**Bedeutung**
Dunkel	Keine Verbindung
Dauerlicht	Datenverbindung, Passwort OK, IEC-Kommunikation
Blinken langsam	Verbindungsaufbau eingeleitet durch ALU
Blinken schnell	Verbindungsaufbau eingeleitet durch Gegenstelle (RING)
2 mal Blinken (500ms) - Pause	Modem Initialisierungsproblem. In diesem Fall sollten die Einträge AT_INIT, AT_INI2, AT_INI3 in der Datei SVC_COMn.INI überprüft werden.
3 mal Blinken (300ms) - Pause	Modem nicht mehr bereit

Anmerkung: Die Funktion MODEM_LED steht erst ab ALU-Firmwareversion 1.91 zur Verfügung.

6.4.4 Parameterdatei für die Datenpunktdefinition bearbeiten (SVC_RTUn.INI)

 **Hinweis:** Ab Firmware-Version 1.70 werden geänderte Dateinamen verwendet (SVCRTUn.INI)

[RTU_I10X] ASDU= 3 OKT_ASDU= 2 OKT_HERK=TRUE OKT_AINF= 3	 ASDU = Stationsadresse Anzahl Oktett der Adressbyte Übertragungsursache mit Herkunftsadresse Anzahl Oktett der Informationsobjektnummer
APDU_LEN=253 BLOCK_OBJEKT= TRUE BLOCK_ELEMENT= TRUE	Maximale Länge der Blocktelegramme Blockung von Informationsobjekten erlaubt TRUE / FALSE Blockung von Informationselementen erlaubt TRUE / FALSE
IEC_CCI_TYP= O	Zählwertabfrage Abfrage Betriebsart [C | D | O]
RTU_COM_STOP=TRUE	Verhalten bei STOP des Anwenderprogramms, TRUE= Kommunikation stoppt, FALSE= läuft weiter
EXEC_CMDNENABLE = TRUE EXEC_CMDNTIME = 10 EXEC_STIME= 3 EXEC_LTIME= 50	Befehle mit Qualitätskennung QU=0 sind erlaubt Befehlsausführungszeit für Befehle ohne zusätzliche Festlegung (QU=0) Kurze Befehlsausführungszeit (n * 100 ms) Lange Befehlsausführungszeit (n * 100 ms)
1:AP_NMB= 3600 1:AP_WARN= 2500	Anzahl Telegramme im Archiv für PVs mit Zeitmarke (0 bis 3600) Überlaufwarnung des Archivpuffers (führt bei AWD-Betrieb zum Verbin-dungsaufbau).
OOFS_SP=100 OOFS_SP_TM=100 OOFS_DP=200 OOFS_DP_TM=200 OOFS_BO=300 OOFS_BO_TM=300 OOFS_ME=400 OOFS_ME_TM=400 OOFS_IT=500 OOFS_IT_TM=500	Objektnummern - Offset Einzelmeldungen (Single Point) Objektnummern - Offset Einzelmeldungen (Single Point) mit Zeitmarke Objektnummern - Offset für Doppelmeldungen (Double Point) Objektnummern - Offset für Doppelmeldungen (Double Point) mit Zeitm. Objektnummern - Offset Bitstring Objektnummern - Offset Bitstring mit Zeitmarke Objektnummern - Offset Messwerte Objektnummern - Offset Messwerte mit Zeitmarke Objektnummern - Offset Zählwerte Objektnummern - Offset Zählwerte mit Zeitmarke
OOFS_SC= 600 OOFS_DC=700 OOFS_SE=800 OOFS_CBO=900	Objektnummern – Offset Einzelbefehl (Single Command) Objektnummern - Offset Doppelbefehl (Double Command) Objektnummern - Offset Sollwert normiert (Set-point command) Objektnummern - Offset Bitstring (Bitstring of 32 Bit))



# interne Anweisung für die Varianten (Datenverarbeitung) # Datenvariante Var1	
1:VAR_XX_ABF= TRUE 1:VAR_XX_SPO= TRUE 1:VAR_XX_AWD= FALSE 1:VAR_XX_AP= 1 1:VAR_XX_GRP=1	Abfrage aktiv, Generalabfrage bzw. Zählerabfrage Spontan Übertragung freigeben 1= bei AWD Betreib löst Änderung Verbindungsaufbau aus Zyklisch in Archivpuffer 0 = nein, Archiv 1…3 Wert wird bei GA mit Gruppenabfrage 1 übertragen
# interne Anweisung für die Variante1 nur für Doppelmeldungen(DP)	
1:VAR_DP_TS=100	Störstellungsunterdrückungszeit, 0 = keine Überwachung, n * 10 ms
# interne Anweisung für die Variante1 nur für Bitstring(BO)	
1:VAR_BO_NMB=32	Anzahl Prozesspunkte, 1…32 pro Bitstring
# interne Anweisung für die Variante1 nur für Zählwerte(IT)	
1:VAR_IT_IMPS= 0	Impulsschwelle (threshold) 0 bis 65535 (def. = 0, aus)
# interne Anweisung für die Variante1 nur für Zählwerte (IT) und Messwerte (ME)	
1:VAR_ITME_ZYK = 5	Differenz-/Zykluszeit= n*Minute, 0 bis 3600, 0 = aus
# interne Anweisung für die Variante 1nur für Messwerte (ME)	
1:VAR_ME_AZI=512	Abweichungszeitintegral, AZI = 0… 32760 / 0x7FF8 12 Bit Messwert + VZ, linksbündig (AZI = 8 entspricht der Änderung von einem Digit am Eingang)
1:VAR_ME_OV=32000	Overrange (OV) von 0 bis 32767 (0x7FFF)
# interne Anweisung für die Varianten (Datenverarbeitung) # Datenvariante Var2	
2:VAR_XX_ABF= TRUE 2:VAR_XX_SPO= TRUE 2:VAR_XX_AP= 0 2:VAR_XX_GRP=2	Abfrage aktiv, Generalabfrage bzw. Zählerabfrage Spontan Zyklisch in Archivpuffer 0 = nein, Archiv 1…3 Wert wird bei GA mit Gruppenabfrage 2 übertragen
# interne Anweisung für die Variante2 nur für Zählwerte(IT)	
2:VAR_IT_IMPS= 20	Impulsschwelle (threshold) 0 bis 65535 (def. = 0, aus)
# interne Anweisung für die Variante2 nur für Zählwerte (IT) und Messwerte (ME)	
2:VAR_ITME_ZYK = 0	Differenz-/Zykluszeit= n*Minute, 0 bis 3600, 0 = aus
# interne Anweisung für die Variante2 nur für Messwerte (ME)	
2:VAR_ME_AZI=256	Abweichungszeitintegral, AZI = 0… 32760 / 0x7FF8 12 Bit Messwert + VZ, linksbündig (AZI = 8 entspricht der Änderung von einem Digit am Eingang)
2:VAR_ME_OV=32000	Overrange (OV) von 0 bis 32767 (0x7FFF)
#interne Anweisung für die Varianten (Datenverarbeitung)	

# Datenvariante Var3	
3:VAR_XX_ABF= TRUE 3:VAR_XX_SPO= FALSE 3:VAR_XX_AP= 1	Abfrage aktiv, Generalabfrage bzw. Zählerabfrage Keine spontane Übertragung Zyklisch in Archivpuffer 0 = nein, Archiv 1…3
# interne Anweisung für die Variante3 nur für Zählwerte (IT) und Messwerte (ME)	
3:VAR_ITME_ZYK = 5	Differenz-/Zykluszeit= n*Minute, 0 bis 3600, 0 = aus
# interne Anweisung für die Variante3 nur für Messwerte (ME)	
3:VAR_ME_AZI=0	Abweichungszeitintegral, AZI = 0… 32760 / 0x7FF8 12 Bit Messwert + VZ, linksbündig (AZI = 8 entspricht der Änderung von einem Digit am Eingang)
	Es können bis zu 16 verschiedene Verarbeitungsvarianten definiert werden.
# Verknüpfung (Zuordnung) von Prozessdaten in Melderichtung (Monitoring Direction)	
QX0.0:TAG_SP_TM= 1;V1 QX0.0:TAG_SP= 1;V1 QX0.1:TAG_SP= 2;V1 QX0.4:TAG_DP=1;V1 QD4:TAG_BO=1;V2 QW8:TAG_ME= 1;V1; QW10:TAG_ME= 2;V2; QW12:TAG_ME_TM= 3;V1; QW14:TAG_ME_TM= 4;V2; QD16:TAG_IT= 1;V1 QD20:TAG_IT_TM= 2;V1 QD24:TAG_IT= 3;V2 QD28:TAG_IT_TM= 4;V2	Einzelmeldung mit Zeitstempel Einzelmeldungen ohne Zeitstempel Doppelmeldung Bitstring 32 Bit Messwerte ohne Zeitstempel definiert als WORD Messwerte mit Zeitstempel definiert als WORD 32 Bit Zählwerte definiert als DWORD



# Verknüpfung (Zuordnung) von Prozessdaten in Befehlsrichtung (Control Direction)	
IX0.0:TAG_SC= 1;V1 IX0.1:TAG_SC= 2;V1	Im 1. Byte des Übergabefels werden zwei Einzelbefehle eingetragen
IW2:TAG_SE=1;V1 IW4:TAG_SE=2;V1	Im 3. und 4. Byte des Übergabefelds wird ein Sollwert eingetragen Im 5. und 6. Byte des Übergabefelds wird ein Sollwert eingetragen

Erläuterung zu den einzelnen Parametern

ASDU, OKT_ASDU:

Die **A**pplication **S**ervice **D**ata **U**nit ist die Stationsadresse. Die Länge ist nur bei IEC 60870-5-101 variabel, bei IEC 60870-5-104 ist sie fest mit 2 vorgegeben.

OKT_HERK:

Gibt an, ob die Übertragungsursache mit oder ohne Herkunftsadresse übertragen wird. Bei IEC 60870-5-104 ist das Oktett für die Herkunftsadresse immer enthalten, wird von der ALU320 jedoch fest auf NULL gesetzt.

OKT_AINF:

Länge der Adresse des Informationsobjekts. Sie kann 1, 2 oder 3 Byte groß sein. Die Länge ist nur bei IEC 60870-5-101 variabel, bei IEC 60870-5-104 ist sie fest mit 3 vorgegeben.

APDU_LEN= 30

Die maximale Länge der APDU (Protokolldateneinheit der Anwenderschicht = Telegramm) kann vorgegeben werden. Bei IEC-870-5-104 ist die maximale Läge 253 Byte (255 abzüglich Start- und Längenoktett).

BLOCK_OBJEKT

Dieser Schalter gibt an, ob mehrere PVs als geblockte Informationsobjekte übertragen werden dürfen.

Defaulteinstellung = TRUE, da hierdurch eine optimierte Übertragung der PVs insbesondere bei GA möglich ist. Der Schalter sollte nur auf FALSE gestellt werden, wenn das übergeordnete System diese Art von Blocktelegrammen nicht verarbeiten kann.

BLOCK_ELEMENT

Dieser Schalter gibt an, ob mehrere PVs als geblockte Informationselemente übertragen werden dürfen.

Defaulteinstellung = TRUE, da hierdurch eine optimierte Übertragung der PVs insbesondere bei GA möglich ist. Der Schalter sollte nur auf FALSE gestellt werden, wenn das übergeordnete System diese Art von Blocktelegrammen nicht verarbeiten kann.

IEC_CCI_TYP= O

Die Norm sieht für die Übertragung von Zählwerten 4 Verfahren vor. Betriebsart A, B, C und D. Die Betriebsart A kann durch den Parameter VAR_ITME_ZYK bei der Definition der Verarbeitungsvarianten realisiert werden.

Bei den Betriebsarten B, C und D wird ein Zählerabfragebefehl (CI = Counter Interrogation) vom übergeordneten System geschickt. Die ALU 320 unterstützt nur die Betriebsarten C und D, sowie ein OHP spezifisches Verfahren (O) bei dem die Zählwerte sofort nach einem Umspeicherbefehle (Zählerabfragebefehl mit FRZ=1, RQT=1…5) mit der Ursache 37 … 41 übertragen werden. Dies ist eine Abwandlung der Betriebsart C (eine Zählerabfrage FRZ=0, RQT=1…5 wird hier nicht gesendet).

RTU_COM_STOP

Über diesen Schalter wird vorgegeben, ob die Kommunikation zum Übergeordneten System bei AWP-Stopp (AWP = Anwenderprogramm) ebenfalls gestoppt werden soll oder nicht.

FALSE = Kommunikationstreiber laufen weiter und antworten auf Pollaufrufe mit Kurzantwort oder E5. Alle Projektierten PVs werden bei AWP-Stopp mit gesetztem Invalid-Bit übertragen. Wird das AWP wieder gestartet werden alle Invalid-Bits zurückgesetzt und die PVs werden übertragen.

RTU_TIME_SET

0 = Zeit darf über IEC gestellt werden (default)
1 = Zeit darf nie über IEC gestellt werden
2 = Zeit darf nur über IEC gestellt werden, wenn aktuelle Zeit IV (ungültig en: invalid) ist

RTU_TIME_ACK_NEG

FALSE = IEC Zeittelegramm wird mit positiver Response bestätigt (default)
TRUE = IEC Zeittelegramm wird mit negativer Response bestätigt

Dieser Parameter ist nur gültig, wenn RTU_TIME_SET ungleich NULL ist!

RTU_TIME_SYNC_ENABLE

Eingabe in Minuten (1…65535)
Delta-Zeitangabe in Minuten, seit dem letzten Stellen der Uhr. Nach Ablauf darf die Uhr über IEC-Telegramm gestellt werden.
Dieser Parameter ist nur gültig, wenn RTU_TIME_SET = 2 ist!

EXEC_CMDNENABLE, EXEC_CMDNTIME:

Wenn der Parameter EXEC_CMDNENABLE = TRUE gesetzt wird, werden Befehle mit der Qualitätskennung OU = 0 (ohne zusätzliche Festlegung) entgegen genommen und mit der in Parameter EXEC_CMDNTIME festgelegten Zeit ausgegeben. Wird der Parameter EXEC_CMDNENABLE = FALSE gesetzt, werden Befehle mit QU=0 negativ bestätigt und nicht ausgeführt.

Normalerweise wird diese Einstellung verwendet, wenn die Befehlsausführungszeit in der Station für jeden Befehl individuell festgelegt wird. Sollte dies im Anwenderprogramm so realisiert sein, ist zu beachten, dass die ALU320 die "Beendigung der Aktivierung" (Ursache 10) immer nach Ablauf der durch EXEC_CMDNTIME vorgegebenen Zeit sendet.

EXEC_STIME, EXEC_LTIME:

Hier wird die kurze und lange Befehlsausgabezeit vorgegeben. Mit welcher Zeit der Befehle ausgegeben wird, wird durch die Befehlskennung im Telegramm von der Zentrale festgelegt. Zeitbasis: n * 100 ms.

?:AP_NMB, ?:AP_WARN:

Es wird die Anzahl von Telegrammen die der Archivpuffer fassen soll vorgegeben (AP_NMB).

Der Parameter ?:AP_WARN ist nur bei AWD-Betrieb aktiv. Er gibt an, bei welchem Füllstand des Puffers von der Unterstation automatisch eine Verbindung zum übergeordneten System eingeleitet wird.

Es können maximal 3600 Telegramme gepuffert werden. Diese Anzahl kann auf bis zu 3 Archive verteilt werden.



OOFS_??:

Über diese Parameter können Sie pro Datentyp einen Offset vorgeben, der auf die bei der Definition der Eingänge angegebene Objektnummer aufaddiert wird. Die Verwendung der Offsets ermöglicht es **alle** Objektnummern eines Datentyps in einen anderen Bereich zu verschieben.

Im obigen Beispiel werden die ersten 2 Bit im ersten Übergabebyte als Einzelmeldungen mit den Objektnummern 101 und 102 übertragen da der Parameter OOFS_SP mit 100 angegeben ist.

Bit 4 und 5 werden als Doppelmeldung mit der Objektnummer 201 übertragen.

Byte 4..7 des Übergabeblocks (definiert als DWORD) werden als Bitstring mit der Objektnummer 301 übertragen.

Die 4 Messwerte, die ab Byte 8 (definiert als WORD) übergeben werden, werden jeweils mit und ohne Zeitstempel mit den Objektnummern 401, 402, 403 und 404 übertragen.

Die 4 Zählwerte, die ab Byte 16 (definiert als DWORD) übergeben werden mit den Objektnummern 501, 502, 503 und 504 übertragen.



Hinweis: Es wird empfohlen die Offsets für PVs mit und ohne Zeitstempel gleich einzustellen und die Eingänge fortlaufend durchzunummerieren. Dadurch können bei einer Generalabfrage die Daten optimiert übertragen werden.

Verarbeitungsvarianten:

Es können bis zu 16 unterschiedliche Verarbeitungsvarianten (1:VAR…, 2:VAR… usw.) definiert werden. Diese werden anschließend den einzelnen Eingängen zugeordnet. Es gibt Parameter die für alle Datentypen gültig sind (gekennzeichnet mit XX), andere beziehen sich nur auf bestimmte Datentypen (z.B. VAR_IT_IMPS gültig nur bei Zählwerten).

?:VAR_XX_ABF

TRUE = Prozessvariable wird bei Generalabfrage (global) übertragen.

?: VAR_XX_GRP=1

Über diesen Parameter kann die Generalabfrage als Gruppenabfrage (Gruppe 1…16) ausgeführt werden.

?:VAR_XX_SPO

TRUE = Prozessvariable wird bei Änderung spontan übertragen

?:VAR_XX_AWD

TRUE = Einleiten des Verbindungsaufbau zur Zentrale bei Änderung der Prozessvariablen

?:VAR_XX_AP

Prozessvariable zyklisch ins Archiv Nr. 1, 2 oder 3 schreiben, 0 = inaktiv

?:VAR_BO_NMB

Über diesen Parameter geben Sie an, wie viele hintereinander liegende Bit im Übergabefeld in einem Bitstring zusammengefasst werden. Bereich: 1…32

?:VAR_IT_IMPS

Impulsschwelle im Bereich 0 bis 65535. Bei Erreichen der projektierten Impulsschwelle wird der Zählwert spontan übertragen (Ursache 3). Einstellung 0 bedeutet, dass der Zählwert nur bei Zählerabfrage oder, falls projektiert, zyklisch übertragen wird.

?:VAR_ITME_ZYK

Zeitintervall für die zyklische Übertragung von Zählwerten und Messwerten. Die Werte werden je nach Einstellung von Parameter OFFS_IT / OFF_ME oder OOFS_IT_TM / OOFS_ME_TM mit oder ohne Zeitmarke und der Ursache 3 (spontan) übertragen.

Angabe in Minuten von 1 bis 3600. Einstellung 0 = keine zyklische Übertragung.

?:VAR_ME_AZI

Einstellen des Abweichungszeitintegrals. Die Messwertänderung wird ab dem Zeitpunkt der letzten Übertragung in 1 Sekundenabständen erfasst und vorzeichenrichtig aufaddiert. Erreicht das Ergebnis dieser Addition den eingestellten AZI, wird der Messwert mit der Ursache 3 (spontan) übertragen. Durch dieses Verfahren wird die Messwertübertragung beruhigt. Langsam ansteigende Messwerte werden verzögert übertragen, schnell ansteigende Werte werden früher übertragen.

Bei Einstellung AZI = 0 wird jede Messwertänderung übertragen falls der Parameter VAR_XX_SPO=TRUE ist. Beachten Sie hierbei, dass Messwerteeingänge generell schwanken.

?: VAR_ME_OV

Hier kann die Grenze für den Überlauf (en:Overrange) projektiert werden. Überschreitet der Messwert den hier angegebenen Wert, wird das Überlaufbit in der Qualitätskennung gesetzt. Einstellung 0 = keine Überwachung.

Messwerte werden 1:1 aus dem Übergabefeld übernommen und ins Telegramm eingetragen. Es ist durch das Anwenderprogramm sicher zu stellen, dass die Messwerte im durch die Norm festgelegten Format übergeben werden.

Definition der Übergabeblöcke:

Syntax: Referenz Übergabefeld:IEC-Typ=Objektnummer;Verarbeitungsvariante

Die Referenz auf das Übergabefeld kann als QX?.? (BIT) QB (BYTE), QW (WORD) oder QD (DWORD) angegeben werden. In Befehlsrichtung werden IX?.?, IB, IW und ID verwendet.

Die Referenz ist **relativ** und bezieht sich immer auf das für die Service-Routine definierte Übergabefeld. Wurde in der I/O Konfiguration von MULTIPROG der Output für den Service z.B. von %QB501 bis %QB628 definiert, legt QB0 also die Objektdefinition/Verarbeitung der in %QB501 eingetragenen Daten fest.

Maximale Länge der Übergabeblöcke:

Befehlsrichtung	512 Byte
Melderichtung	1024 Byte

Meldungen werden über QX?.? definiert. Bei Doppelmeldungen wird immer das erste der beiden aufeinander folgenden Bit als Referenz angegeben.

Messwerte (16 Bit) werden über QW definiert

Bitstrings und Zählwerte (32 Bit) werden über QD definiert.

Befehle werden über IX?.? definiert wobei bei Doppelbefehlen jeweils das erste Bit der beiden aufeinander folgenden Bit als Referenz angegeben wird.

Sollwerte (16 Bit) werden über IW definiert

Bitstrings (32 Bit) werden über ID definiert.



Beispiele

:

QX0.0:TAG_SP_TM= 1;V1

QX0.0:TAG_SP= 1;V1

Das 1. Bit im Übergabeblock wird als Einzelmeldung (en: **S**ingle **P**oint) sowohl mit, als auch ohne Zeitstempel übertragen. Für die Übertragung gelten die in Variante 1 projektierten Parameter.

QD16:TAG_IT= 1;V1

Der in Byte 16 bis 19 übergebenen Zählwert wird mit den in Verarbeitungsvariante 2 festgelegten Parametern übertragen. Im Beispiel immer nach 5 Minuten und bei Zählerabfrage.

QD28:TAG_IT= 4;V2

Der in Byte 28 bis 31 übergebenen Zählwert wird mit den in Verarbeitungsvariante 2 festgelegten Parametern übertragen. Im Beispiel immer nach 5 Zählimpulsen mit Zeitstempel und bei Zählerabfrage.

TAGs der IEC-Typen:

TAG_SP	Einzelmeldungen (en: **S**ingle **P**oint Information)
TAG_SP_TM	Einzelmeldungen mit Zeitmarke (en: **S**ingle **P**oint Information)
TAG_DP	Doppelmeldungen (en: **D**ouble **P**oint Information)
TAG_DP_TM	Doppelmeldungen mit Zeitmarke (en: **D**ouble **P**oint Information)
TAG_IT	Zählwert (en: **I**ntegrated **T**otal)
TAG_IT_TM	Zählwert mit Zeitmarke (en: **I**ntegrated **T**otal)
TAG_ME	Messwert normiert (**Me**asured Value normalized)
TAG_ME_TM	Messwert normiert mit Zeitmarke (**Me**asured Value normalized)
TAG_BO	Bitmuster von 32 Bit (en: **B**itstring **O**f 32 Bit)
TAG_BO_TM	Bitmuster von 32 Bit mit Zeitmarke (en: **B**itstring **O**f 32 Bit)
TAG_SC	Einzelbefehl (en: **S**ingle **C**ommand)
TAG_DC	Doppelbefehl (en: **D**ouble **C**ommand)
TAG_SE	Sollwert-Stellbefehl normalisiert (en: Set-point command normalized value)
TAG_CBO	Bitmuster von 32 Bit Befehlsrichtung (en: **B**itstring **O**f 32 Bit command direktion)

6.5 Projektierung der Initialisierungsdateien für Modbus RS232-Slave oder TCP-Server

Die Datei MODBUS_SERVER.INI ist im Verzeichnis PCOS_OHP abgelegt und gilt sowohl für den Modbus RS232-Slave wie auch für den Modbus TCP-Server.

 **Hinweis:** Ab Firmware-Version 1.70 werden geänderte Dateinamen verwendet (MBSRV.INI)

```
#=========================================
# MODBUS SERVER / SLAVE Konfigurationsdatei
# MODBUS SERVER / SLAVE configuration file
#=========================================

#---------------------------------------------------
# Interface 1 - SERIAL - Seriell
#---------------------------------------------------

# Communicationparameter: Modbus Slave Serial Interface - Service 1
```

[IF1-COM-MODBUS-SSERIAL] PORT= 2 BAUDRATE= 9600 PARITY= EVEN DATA=8 STOPBIT=1 PAUSE=10 BUS_TOUT= 2000 ASCII= FALSE SLAVE_ADR= 1	 # PORT [xx], 0 ... aus # Stoppbit (1= default) # mindesten Pausenzeit [ms] zwischen Request u. Response # Busueberwachung (Slave) in n*[ms], def. (2000), Bereich 1000 bis 2000 # Protokoll ASCII FALSE/TRUE; (def.FALSE -> RTU)

Serverparamter für Modbus Slave Serial interface - Service 1

[IF1-SERV-MODBUS] REF_BASEINDEX= 0 STATUS_OFFSET= 300 STATUS_LEN= 6	# 0/1 default (0) # 2, 4 oder 6 Byte lang
# 0x COILS - Read/Write - 1 bit access REF0_OFFSET= 0 REF0_LEN=50	 # REF0: Offset in Byte-Modbus Server # Länge in Byte (50 x 8 COILS)
# 1x Discret Input - Read - 1 bit access REF1_OFFSET= 50 REF1_LEN=50	 # REF1: Offset in Byte-Modbus Server # Länge in Byte (50 x 8 Discrete Inputs)
# 3x Input Register - Read - 16 bit access REF3_OFFSET= 100 REF3_LEN=100	 # REF3: Offset in Byte-Modbus Server # Länge in Byte (100 entsp. 50 Input Register)
# 4x Holding Register - Read/Write - 16 bit access REF4_OFFSET= 20 REF4_LEN=100	 # REF4: Offset in Byte-Modbus Server # Länge in Byte (100 entsp. 50 Holding Register)

```
#------------------------------------------------------------------------------
# Interface 2 - Modbus TCP/IP Server (MODBUS-STCP_IP)
#------------------------------------------------------------------------------

# Communicationparameter: Modbus - STCP_IP interface 2
```

[IF2-COM-MODBUS-STCP_IP] //IP_ADDR= 192.168.0.101	 #IP-Adresse des Teilnehmers



//IP_ADDR= 192.168.0.102
//IP_ADDR= 192.168.0.103
//IP_ADDR= 192.168.0.104
	#IP-Adresse des Teilnehmers
	#IP-Adresse des Teilnehmers
	#IP-Adresse des Teilnehmers

# Serverparamter für Modbus TCP interface 2	
[IF2-SERV-MODBUS] REF_BASEINDEX= 0 STATUS_OFFSET= 300 STATUS_LEN= 6	# 0/1 default (0) # 2, 4 oder 6 Byte lang
# 0x COILS - Read/Write - 1 bit access REF0_OFFSET= 0 REF0_LEN=50	# REF0: Offset in Byte-Modbus Server # Länge in Byte (50 x 8 COILS)
# 1x Discret Input - Read - 1 bit access REF1_OFFSET= 50 REF1_LEN=50	# REF1: Offset in Byte-Modbus Server # Länge in Byte (50 x 8 Discrete Inputs)
# 3x Input Register - Read - 16 bit access REF3_OFFSET= 100 REF3_LEN=100	# REF3: Offset in Byte-Modbus Server # Länge in Byte (100 entsp. 50 Input Register)
# 4x Holding Register - Read/Write - 16 bit access REF4_OFFSET= 20 REF4_LEN=100	# REF4: Offset in Byte-Modbus Server # Länge in Byte (100 entsp. 50 Holding Register)

6.5.1 Erläuterung zu den einzelnen Parametern

Mit [IF1…], [IF2…] , [IF3…] usw. wird angegeben auf welches Interface sich die darauf folgenden Parameter beziehen.

[IF1-COM-MODBUS-SSERIAL]

[IF-1-COM-MODBUS-SSERIAL] gibt an, dass diese Einstellungen für das Interface1 mit einer serielle Modbus-Kopplung gelten.

PORT

Hier geben Sie die COM-Schnittstelle (1 oder 2) an, die Sie für die Kommunikation mit Modbus RS232-Slave verwenden wollen.

BAUDRATE

Die Baudrate kann zwischen 600 und 9600 Baud eingestellt werden.

PARITY, DATA, STOPBIT

Bei der Parität kann zwischen EVEN / ODD bzw. GERADE / UNGERADE gewählt werden. Die Anzahl der Datenbit muss immer 8 sein, die Anzahl Stoppbit immer 1.

PAUSE

Pausenzeit in ms zwischen Anfrage und Antwort. Über diesen Parameter kann das Antworttelegramm verzögert werden, falls der Master nicht in der Lage ist, eine Antwort sofort entgegen zu nehmen.

BUS_TOUT

Busüberwachung in ms. Bereich 1000 bis 2000, Defaulteinstellung 2000 (2 Sekunden).

Der Slave prüft, ob innerhalb der eingestellten Überwachungszeit auf dem Bus Telegramme übertragen werden. Ist dies nicht der Fall, wird der Status "Kommunikationsstörung" gesetzt.



ASCII
Schaler für Modbus-ASCII oder Modbus-RTU. Derzeit ist **nur** Modbus-RTU möglich, Einstellung ASCII=FALSE.

SLAVE_ADR
Hier stellen Sie die Slave- bzw. Node- Adresse ein.

[IF1-SERV-MODBUS]

Die folgenden Anweisungen und Parameter gelten für das Interface1 und definieren den Bereich, der über Modbus gelesen und geschrieben werden kann.

REF_BASEINDEX= 0

Hier stellen Sie ein, ob der Modbus-Master das 1. Coil bzw. Register mit 0 oder 1 adressiert.
Defaulteinstellung ist 0.

STATUS_OFFSET, STATUS_LEN

Über diese beiden Parameter wird festgelegt, ab welcher MB3.???? Adresse der Modbus Status abgelegt wird und wie viele Byte für den Status angezeigt werden sollen. Die Länge kann mit 2, 4 oder 6 Byte vorgegeben werden.

 Weiterführende Informationen siehe Kapitel 5.5.5.

Der für die Modbus Kommunikation vorgesehene Shared Memory Bereich kann aufgeteilt werden in Bit- und Wort-Bereiche. Hierfür kann jeweils ein Offset und die Länge des Bereichs angegeben werden. Diese Bereiche können dann mit den entsprechenden Funktionscodes gelesen bzw. beschrieben werden.

In Anlehnung an andere Modbus Projektierungen wurden diese Bereiche wie folgt definiert:

0er Referenz = Ausgangs- oder Merkerbits
1er Referenzen = Eingangsbit
3er Referenzen = Eingangswort
4er Referenzen = Ausgangs- oder Merkerwort

REF0_OFFSET, REF0_LEN

Dieser Bereich kann mit den folgenden Funktionscodes verarbeitet werden

FC 1	Lese Bit	Read Coils
FC 5	Schreibe einzelnes Bit	Write Single Coil
FC 15	Schreibe Bitbereich	Write Multiple Coils

REF1_OFFSET, REF1_LEN

Dieser Bereich kann mit den folgenden Funktionscodes verarbeitet werden

FC 2	Lese Eingang	Read Discrete Inputs

REF3_OFFSET, REF3_LEN

Dieser Bereich kann mit den folgenden Funktionscodes verarbeitet werden

FC 4	Lese Eingangswort	Read Input Register



REF4_OFFSET, REF4_LEN

Dieser Bereich kann mit den folgenden Funktionscodes verarbeitet werden

FC 3	Lese Register	Read Holding Register
FC 6	Schreibe einzelnes Register	Write Single Register
FC 16	Schreibe Registerbereich	Write Multiple Register
FC 23	Lese/Schreibe Registerbereich	Read/Write Multiple Register

Die Bereiche können auch überlappend projektiert werden.

Mit der Einstellung

REF0_OFFSET= 0
REF0_LEN=50

REF4_OFFSET= 0
REF4_LEN=50

können die ersten 50 Byte des Shared Memory sowohl mit den Funktionscodes FC1, FC5 und FC15 wie auch mit den Funktionscodes FC3, FC6, FC16 und FC23 gelesen bzw. geschrieben werden.

[IF2- COM-MODBUS-STCP_IP]

[IF-2-COM-MODBUS-STCP_IP] gibt an, dass diese Einstellungen für das Interface2 mit einer Modbus TCP-Server Kopplung gelten.

IP_ADDR= ?????
Falls die Modbus TCP-Kopplung nur für bestimmte Clients freigegeben werden soll, können deren IP-Adressen angegeben werden. Es können bis zu 4 Verbindungen gleichzeitig aufgebaut werden. Wird keine IP-Adresse angegeben, erhalten die ersten 4 Clients zugriff.

[IF2-SERV-MODBUS]

Die folgenden Anweisungen und Parameter gelten für das Interface2 und definieren den Bereich, der über Modbus gelesen und geschrieben werden kann.

Beschreibung für Baseindex, Status und Referenzen siehe oben.

Mit der Beispielprojektierung kann der gleiche Speicherbereich mit einer seriellen Modbus wie auch mit einer Modbus TCP/IP Kopplung abgefragt und beschrieben werden, wenn beide Interfaces in der Datei PCOS_A320.INI freigegeben werden.

Den beiden Kopplungen können jedoch auch getrennte Bereiche zugewiesen werden. Dies ist dann entsprechen im Shared-Memory und bei der Definition der Referenzen zu berücksichtigen.

7 Firmwarebibliotheken

7.1 Allgemeines

Für die Multiprog-Version 4.6 wurde die Bibliotheken ALUX2X_OHP_??? entwickelt. Mit der Umstellung auf Multiprog 4.8 und der Integration der Funktion "Änderungen senden" (Online-Ändern) wurden diese Firmwarebibliotheken überarbeitet. Diese neuen Bibliothek haben den Namen ALUX2X_OHP4_???.

Um für ältere Projekte kompatibel zu bleiben, werden bei der Installation der OHP-Add-Ons beide Varianten installiert. Die neuen Projekte-Vorlagen beinhalten jedoch nur die neue Varianten ALUX2X_OHP4_???.

Wenn Sie mit einem älteren Projekt die Leistung "Änderungen senden" nutzen möchten, müssen Sie die Bibliotheken austauschen. Klicken Sie hierzu im Projektbaum-Menü den Eintrag *Bibliotheken* mit der rechten Maustaste an. Danach können Sie über *Einfügen – Firmware-Bibliothek* die neuen Bibliotheken einbinden.



Pfad zu den FW-Bibliotheken: [Lw]:\Programme\OHP\Multiprog_48\Multiprog\PLC\FW_LIB\

Die alten Bibliotheken können durch Anwahl mit der Maus und Betätigen der Taste *Entf* gelöscht werden.

Voraussetzung für das Online-Ändern ist, dass im Projekt bereits die neuen Ressourcen verwendet werden, und die ALU 320 eine Firmware ab Version 3.00 enthält.

Weiterführende Informationen hierzu siehe **Teil II Projektierung und Programmierung,** Kapitel 8.1.

Nach Austausch der Bibliotheken muss das Projekt einmal komplett neu übersetzt (*Code – Projekt neu erzeugen*) und gesendet werden Anschließend steht die Funktion "Änderungen senden" zur Verfügung.

OHP stellt folgende Bibliotheken zur Verfügung:

Für PROCONOS 3

ALUX20_OHP_SCOM

ALUX20_OHP_SEAB

ALUX20_OHP_SYS

Für PROCONOS 4 (Online-Ändern)

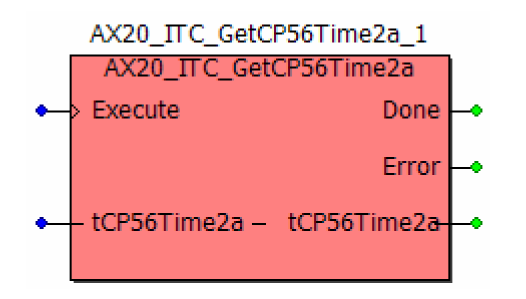
ALUX20_OHP4_SCOM	(Funktionen zu freien Kommunikation über die RS232-Schnittstelle)
ALUX20_OHP4_SEAB	(Funktionen für die Kopplung über IKOS)
ALUX20_OHP4_SYS	(Diverse allgemeine Funktionen)
ALUX20_OHP4_SYS_SA	(Funktionen zur Störungsanalyse und zur Auswertung für DEA – Ankopplung)

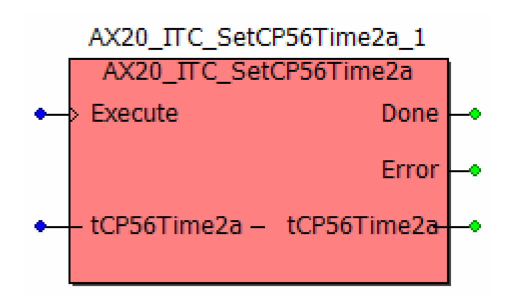


7.2 Bibliothek ALUX20_OHP4_SYS

7.2.1 Auslesen und Stellen der ALU320 Systemzeit im Anwenderprogramm

Funktionsbaustein AX20_ITC_GetCP56Time2a und AX20_ITC_SetCP56Time2a





Eingangsvariable (VAR_IN):
Execute : BOOL Freigabeflag für den Funktionsbaustein, 1 = ausführen

Ausgangsvariable (VAR_OUT):
Done : BOOL 1 = Funktion ausgeführt
Error : BOOL 1 = Funktion fehlerhaft

Ein-/Ausgangsvariable (VAR_IN_OUT):
tCP56Time2a : 7 Byte Systemzeit (Datum/Uhrzeit)

Diese Systemzeit wird nach IEC 60870-5-4, CP56Time2a in 7 Byte wie folgt dargestellt:

<table>
<tr><td colspan="8">**2^7 Millisekunden 2^0**</td></tr>
<tr><td colspan="8">2^{15} **0…59999** 2^8</td></tr>
<tr><td>IV</td><td>Res</td><td colspan="6">2^5 Minute 0…59 2^0</td></tr>
<tr><td>S</td><td colspan="2">Res</td><td colspan="5">2^4 Stunden 0…23 2^0</td></tr>
<tr><td colspan="3">2^7 WT 1..7 2^5</td><td colspan="5">2^4 Tag im Monat 1…31 2^0</td></tr>
<tr><td colspan="4">Res</td><td colspan="4">2^3 Monat 1…12 2^0</td></tr>
<tr><td>Res</td><td colspan="7">2^6 Jahre 0…99 2^0</td></tr>
</table>

IV = Ungültig (en: invalid)

IV <0> = gültig
IV <1> = ungültig

S = Sommerzeit

S<0> = Normalzeit
S<1> = Sommerzeit

WT = Wochentag

WT<0> = keine Angabe
WT<1> = Montag
:
WT<7> = Sonntag

In Multiprog kann diese Zeit z.B. in einem Array oder in einer Struktur abgebildet werden.

```
TYPE
   ArrayTime      :      ARRAY [1..7] OF BYTE;
END_TYPE

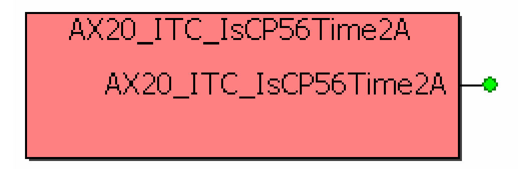
TYPE
IEC_Time:
   STRUCT
         ms:             WORD;
         minute:         BYTE;
         hour:           BYTE;
         day:            BYTE;
         mon:            BYTE;
         year:           BYTE;
   END_STRUCT;
END_TYPE
```

Eine Variable vom Typ IEC_Time oder ArrayTime wird an den Funktionsbaustein AX20_ITC_GetCP56Time2a übergeben um die aktuelle Systemzeit auszulesen, bzw. an den Funktionsbaustein AX20_ITC_SetCP56Time2a um die Zeit zu stellen.

Die OHP-Projekt-Vorlagen beinhalten bereits die Struktur IEC_Time in der Datentyp-Deklaration.

7.2.2 Status der ALU320 Systemzeit lesen

Funktion AX20_ITC_IsCP56Time2A



Achtung: Verwenden Sie keine EN/ENO für die OHP-Funktionen

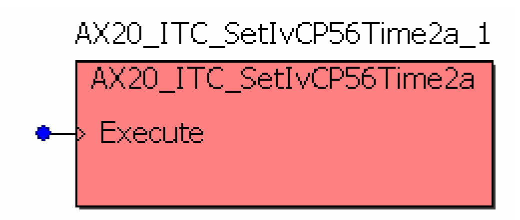
Ausgangsvariable (VAR_OUT):
AX20_ITC_IsCP56Time2A : BOOL Status der Systemzeit

0 = Zeit ist ungültig (invalid)
1 = Zeit ist gültig



7.2.3 Status der ALU320 Systemzeit auf ungültig setzen

Funktionsbaustein AX20_ITC_SetIvCP56Time2A

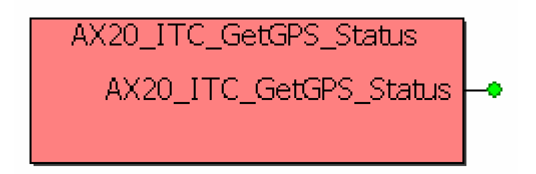


Eingangsvariable (VAR_IN):
Execute : BOOL Freigabeflag für den Funktionsbaustein, 1 = ausführen

Wird dieser Funktionsbaustein ausgeführt, wird die ALU-Systemzeit auf ungültig gesetzt. Dieser Aufruf kann z.B. in einer Systemtask Kalt und/oder Warmstart verwendet werden, um die Systemzeit nach einem Neustart auf ungültig zu setzen, bis sie wieder gezielt gestellt wird (GPS, SNTP).

7.2.4 Status der ALU320 Systemzeit bei GPS-Synchronisation

Funktion AX20_ITC_GetCP56GPS_Status



Achtung: Verwenden Sie keine EN/ENO für die OHP-Funktionen

Ausgangsvariable (VAR_OUT):
AX20_ITC_GetCP56GPS_Status : DWORD Status der Systemzeit bei GPS-Synchronisation

Status in Hex:

16#00000001 GPS-Modul bereit

16#00000003 Uhrzeit gestellt

16#81000001 Uhrzeit kann wegen GPS-Warnung "Uhrzeit ungültig" nicht übernommen werden

16#82000001 Fehler beim Setzen der Uhrzeit

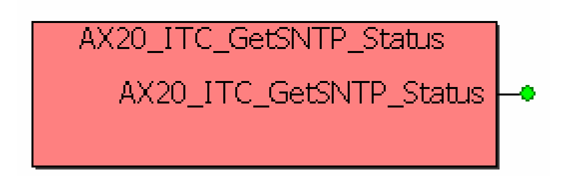
16#84000001 RMC Datensatzfehler

16#88000001 Keine Verbindung zum GPS-Modul



7.2.5 Status der ALU320 Systemzeit bei SNTP-Synchronisation

Funktion AX20_ITC_GetCP56SNTP_Status



Achtung: Verwenden Sie keine EN/ENO für die OHP-Funktionen

Ausgangsvariable (VAR_OUT):
AX20_ITC_GetCP56SNTP_Status : DWORD Status der Systemzeit bei SNTP-Synchronisation

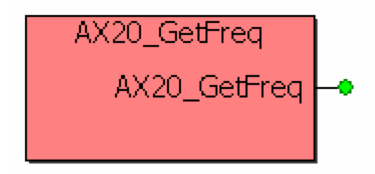
Status in Hex:

16#00000001 Verbindung zum SNTP-Server bereit

16#00000003 Uhrzeit gestellt

7.2.6 Systemtakte

Funktion AX20_ITC_GetFreq



Achtung: Verwenden Sie keine EN/ENO für die OHP-Funktionen

Ausgangsvariable (VAR_OUT):
AX20_ITC_GetFreq : WORD Blinktakte



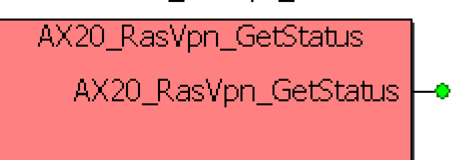
Hinweis: Für die Programmiersprache AWL (en: Instruction List) gibt es einen gesonderten Baustein ILAX20_GetFreqk. Die Funktionalität ist die gleiche, jedoch ist der Baustein mit einem zusätzlichen Execute-Eingang versehen.

Bit	Frequenz	Zeitraster
X0	20 Hz	50 ms
X1	10 Hz	100 ms
X2	5 Hz	200 ms
X3	2,5 Hz	400 ms
X4	1,25 Hz	800 ms
X5		nicht definiert
X6		nicht definiert
X7		nicht definiert
X8	2 Hz	500 ms
X9	1 Hz	1 s
X10	0,5 Hz	2 s
X11	0,25 Hz	4 s
X12	0,125 Hz	8s
X13		nicht definiert
X14		nicht definiert
X15		nicht definiert



7.2.7 Status der GPRS-Verbindung

Funktion AX20_RasVpn_GetStatus



Achtung: Verwenden Sie keine EN/ENO für die OHP-Funktionen

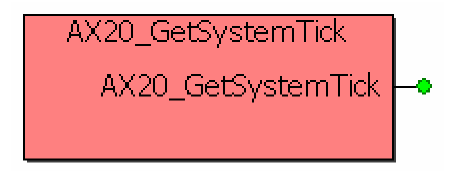
Ausgangsvariable (VAR_OUT):
AX20_RasVpn_GetStatus : DWORD Status der Verbindung

Die Funktion liefert den Status der RAS- und VPN-Verbindung bei Datentransfer über das interne GPRS-Modem.

Status in Hex	**Bedeutung**
16#00000001	Initialisierung läuft
16#00000002	GPRS-Modul bereit
16#00000003	GPRS-Modul bereit - keine Verbindung
16#00000004	GPRS-Modul bereit - Modem Verbindung abgebrochen
16#00000005	GPRS-Modul bereit - VPN Verbindung/Login abgebrochen
16#00000006	GPRS-Modul bereit - VPN-Verbindung kein Login - PPP abgebrochen
16#00000008	Start Modem-Verbindung
16#00000009	Modem Verbindung ok
16#0000000A	Start VPN-Verbindung
16#0000000B	VPN-Tunnel steht
16#0000000C	Zwangstrennung LOGOUT/LOGIN
16#0000000D	Zwangstrennung nach Trafic
16#0000000E	Hardware-Reset ausgelößt
16#0000000F	Software-Reset ausgelößt
16#00000010	Anzahl Verbindungsversuche zum Provider erreicht
16#00000011	Anzahl Verbindungsversuche VPN erreicht

7.2.8 SystemTick

Funktion AX20_GetSystemTick



Achtung: Verwenden Sie keine EN/ENO für die OHP-Funktionen

Ausgangsvariable (VAR_OUT):
AX20_GetSystemTick : DWORD Zeit in Millisekunden

Die Funktion liefert die aktuelle Zeit in Millisekunden seit dem Einschalten der ALU.

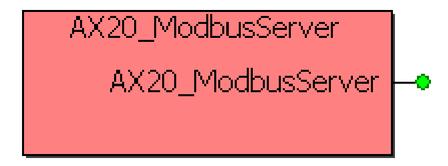
Vorwärtszähler mit Umschlagpunkt. Der Wert wird somit alle 49.71 Tage (2^{32} Millisekunden) wieder auf Null gesetzt.



Hinweis: Für die Programmiersprache AWL (en: Instruction List) gibt es einen gesonderten Baustein ILAX20_GetSystemTick. Die Funktionalität ist die gleiche, jedoch ist der Baustein mit einem zusätzlichen Execute-Eingang versehen.

7.2.9 Modbus-Server

Funktion AX20_ModbusServer



Achtung: Verwenden Sie keine EN/ENO für die OHP-Funktionen

Ausgangsvariable (VAR_OUT):
AX20_ModbusServer : INT Status

Status = 0 = Modbus-Treiber läuft nicht
Status = 1 = Modbus-Treiber läuft

Wenn bei der ALU das Übertragungsprotokoll Modbus-RS232-Slave oder Modbus-TCP-Server benutzt wird, sollte dieser Baustein mindestens einmal in der Default-Task am Anfang oder Ende aufgerufen werden. Der Baustein beantwortet die vom Master/Client gesendeten Abfragen. Dadurch wird sichergestellt, dass die Inhalte der einzelnen Register eines Telegramms aus dem gleichen Programmzyklus stammen. Dies ist insbesondere dann wichtig, wenn Werte berechnet werden und in einem bestimmten Verhältnis zueinander stehen.

Wird der Baustein nicht verwendet, kann es passieren, dass Inhalte eines Telegramms aus verschiedenen Programmzyklen stammen, da der Modbus-Treiber asynchron zum Programmzyklus arbeitet.



Hinweis: Für die Programmiersprache AWL (en: Instruction List) gibt es einen gesonderten Baustein ILAX20_ModbusServer. Die Funktionalität ist die gleiche, jedoch ist der Baustein mit einem zusätzlichen Execute-Eingang versehen.

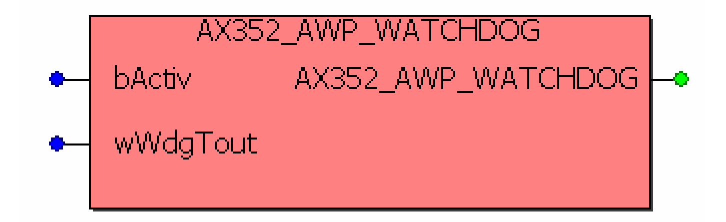


7.3 Bibliothek ALU352 OHP4 AWPWDG

Die Bibliothek steht ab der PCOS Version 3.60 zur Verfügung

7.3.1 Hardware-Watchdog über Anwenderprogramm

Funktionsbaustein AX352_AWP_WATCHDOG



Achtung: Verwenden Sie keine EN/ENO für die OHP-Funktionsbausteine

Eingangsvariable (VAR_IN):
bActive : BOOL Aktivierung der Funktion
wWdgTout : WORD wWdgTout-Zeit kann 0 oder zwischen 100 und 2000 Millisekunden sein

Ausgangsvariable (VAR_OUT):
AX352_AWP_WATCHDOG : WORD Statusinformationen

Status in Hex	Bedeutung
16#0000	Anwender-Task-HW-Watchdog aktiv
16#8000	bActiv= FALSE
16#8001	Hardware Watchdog nicht initialisiert, interne Fehler
16#8002	wWdgTout Zeit ungültig

Der Watchdog wird über bActiv = TRUE und die wWdgTout [ms] aktiviert.

Die wWdgTout-Zeit kann maximal 100 bis 2000 Millisekunden sein.

Der Watchdog kann über bActiv = FALSE und wWdgTout-Zeit= 0ms deaktiviert werden.

Zusätzlich werden im Systemmerker-Bereich noch die folgenden Informationen angezeigt:
Systemmerker %MD1.2928: HW Watchdog Status

Wet in Hex	Bedeutung
16#0000	HW Watchdog nicht aktiv
16#0001	HW Watchdog aktiv
16#0002	HW Watchdog inaktiv – Breakpoint gesetzt
16#0004	HW Watchdog inaktiv – Download Changes, etc.



Hinweis: Der Hardware-Watchdog ist mit dem Software Watchdog einer ProConOS Anwender Task nicht zu verwechseln. Der HW-Watchdog führt nach dem Auslösen zum Reset der SPS.

Die Funktion ALU352_OHP4_AWPWDG aktiviert einen Hardware-Watchdog zur Überwachung einer Anwendertask (z.B. DEFAULT Task) auf der ALU. Der Watchdog wird über bActiv = TRUE und die wWdgTout [ms] aktiviert. Die wWdgTout-Zeit kann zwischen 100 bis 2000 Millisekunden eingestellt werden.

Wenn der Watchdog einmal aktiviert wurde, muss die Watchdoc-Funktion zyklisch (min. 1 x pro Zyklus z.B. in der DEFAULT Task) in kürzeren Abständen aufgerufen werden als in der durch wWdgTout vorgegebenen Zeit. Die wWdgTout-Zeit sollte mindestens das dreifache der aktuellen Zykluszeit betragen, da bei Ablauf der wWdgTout-Zeit die ALU automatisch einen Hardare-Reset durchführt.

Der Watchdog dient dazu das System automatisch neu zu Booten, wenn es z.B. in eine Endlosschleife gelaufen ist oder wenn die SPS steht.

Der Watchdog kann über **bActiv= TRUE und wWdgTout-Zeit= 0ms** deaktiviert werden.

Bei aktiviertem Watchdog führt der Wechsel von bActive =TRUE auf FALSE (wWdgTout-Zeit wird nicht auf 0 gesetzt) ebenfalls zum Hardware-Reset der ALU. Dies kann genutzt werden, um die ALU Ereignisgesteuert zu Booten.

Nach Ablauf der Watchdog Zeit (wWdgTout) versucht der SPS-Prozess-Scheduler die Anwender TASKs zu beenden, d.h. die SPS geht in den STOP Modus die RUN-LED erlischt und die 1. rote LED blinkt. Anschließend führt das System eine START (WARM/KALT) mit HW-Reset des SPS-Systems durch. Sollte das Stoppen der Anwender TASKs nicht erfolgreich sein bzw. der SPS-Prozess-Scheduler kein Zugriff auf das Betriebssystem haben, erfolgt unmittelbar ein Reboot.

Der Anwender braucht bei DEBUG, Online Trace sowie Download Betriebsarten keine Rücksicht auf die eingestellte HW-Watchdog Zeit zu nehmen. Das SPS Laufzeitsystem erkennt selbstständig die Betriebsart und aktualisiert den Watchdog im Hintergrundbetrieb. Sobald der Anwender einen Breakpoint in seinem Anwenderprogramm setzt wird die RUN LED auf den Zustand Blinken gesetzt. Auserdem wird dies über den Systemmerker %MD1.2928 angezeigt.

Um den HW-Watchdog zu testen wird in einem AWP Zyklus der FWB ALU352_OHP4_AWPWDG mit bActiv = TRUE und wWdgTout= 1000 aufgerufen und im nächsten Zyklus wird bActiv= FALSE gesetzt.



8 Tipps und Tricks

8.1 Update eines bestehenden Multiprog Projekts für Online-Ändern

Voraussetzung für das Online-Ändern ist Multiprog 4.8 und eine ALU 320 Firmware ab Version 3.00. Die aktuelle Version kann jeweils von der OHP-Webseite geladen werden.

Wenn Sie mit einem älteren Projekt die Funktion Online-Ändern nutzen möchten, müssen Sie folgende Einstellungen prüfen und gegebenenfalls ändern.

Hardwarestruktur:

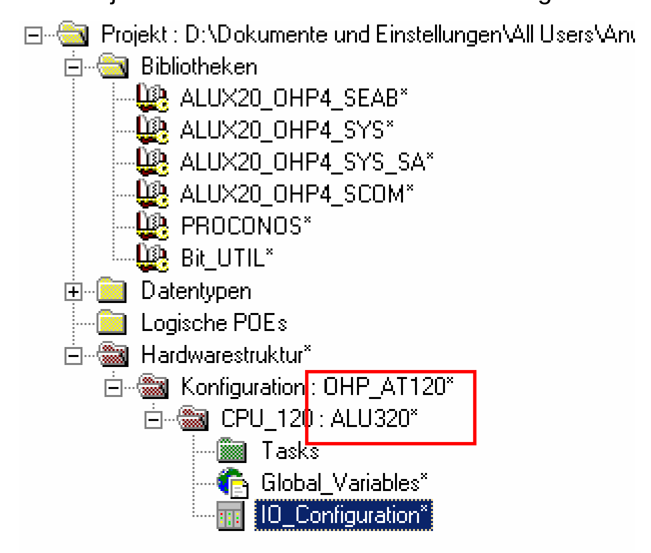
Wenn der SPS-Typ "IPC_33" eingestellt ist, wählen Sie den Eintrag mit der rechten Maustaste an und öffnen Sie den Dialog *Eigenschaften*. Im Register *SPS/Prozessor* wählen Sie dann den *SPS-Typ* "OHP_AT120" aus.

Der CPU-Typ sollte jetzt automatisch auf "ALU320" eingestellt sein. Falls nicht können Sie diesen wieder über Anwahl mit der rechten Maustaste und Aufruf des Dialogs Eigenschaften umstellen.

Bibliotheken:

Wenn Sie in ihrem Projekt Funktionen aus den OHP-Firmware-Bibliotheken genutzt haben, müssen Sie diese ebenfalls updaten. Wie Sie hierbei vorgehen ist im **Teil II Projektierung und Programmierung,** Kapitel 7.1 beschrieben.

Der Projektbaum sollte anschließend wie folgt aussehen.





Hinweis: Nur die rot umrandeten Einträge sind relevant und müssen genau so eingestellt sein. Die Bezeichnungen "Konfiguration" und "CPU_120" können in Ihrem Projekt evtl. anders lauten und müssen nicht geändert werden. Firmware-Bibliotheken müssen nur dann ausgetauscht werden, wenn Sie Funktionen aus diesen benutzt haben.

8.2 Uhrzeit stellen und Diagnose über Telnet

Über eine Telnet-Verbindung können Sie die ALU-Systemzeit stellen oder den Status der ALU-Systemzeit ermitteln.

- Schließen Sie hierfür den Bedien-PC an die Ethernet-Schnittstelle auf der ALU-Unterseite an.
- Geben Sie im Windows Startmenü im Dialog Ausführen *telnet 192.168.0.32* ein (falls Sie die ALU-IP-Adresse geändert haben, geben Sie die geänderte Adresse ein).

In Dialogfenster das anschließend erscheint wird die WindowsCE Version und die Firmware-Version angezeigt.

Es stehen Ihnen jetzt folgende Funktionen zur Verfügung:

- Über *time* wird die aktuelle Uhrzeit angezeigt und kann gestellt werden
- Über *date* wird das aktuelle Datum angezeigt und kann gestellt werden
- *DIAG_TSYNC* zeigt den Status der ALU-Systemzeit an

```
Telnet 192.168.0.52
Welcome to the Windows CE Telnet Service on WinCe 4.2 HCA V1.80B1 90608


Pocket CMD v 4.20
\> DIAG_TSYNC

==========================================================
 DIAG_TSync V1.0 - Diagnose time sync.
==========================================================

 --- PCOS I/O services - SVCn ----------------------------
 NO time synchronisation services activ!

 --- SNTP (NTP0:) services --------------------------------
The SNTP services is not available.

 --- GPS services ----------------------------------------
GPS interface COM(1) Baud= 4800 Refresh= 60 minute
Nmb of event: 0
GPS Status  : Error - GPS modul unreachable, no connect.

 --- PCOS time services  --------------------------------
TimeSyncDelay      : OFF
Status time sync. : INVALID
Last time sync from UNKNOWN

 --- Systeminformation -----------------------------------
Operating in the daylight saving mode.
Time Zone         : W. Europe Standard Time
Standard/Daylight: Bias -60/-60 minute
UTC Time          : 04.09.2008 05:48
Local Time        : 04.09.2008 07:48:59:0

\>
```

PCOS I/O services –SVCn

Wurde die Systemzeit über eine Service-Routine (IEC 60870-5-101 oder 104 Uhrzeitsynchronisation) gestellt, wird dies in dieser Rubrik angezeigt.

SNTP <NTP0:> services

Wurde die Systemzeit über einen Zeit-Server gestellt, wir dies hier angezeigt.

Die Funktion muss in der ALU320.INI frei geschaltet werden.

"SNTP service is not available" bedeutet, dass der Service in der ALU320.INI nicht freigegeben wurde,

GPS services

Zeigt den Status der Synchronisation über den Uhrzeitempfänger GPSRVC an.

Die Funktion muss in der ALU320.INI frei geschaltet werden. Siehe Kapitel 6.1.4.

"GPS service is not available" bedeutet, dass der Service in der ALU320.INI nicht freigegeben wurde,



8.3 ALU-Statusinformationen über Telnet

Über eine Telnet-Verbindung können Sie diverse Diagnosefunktionen aufrufen.

- Schließen Sie hierfür den Bedien-PC an die Ethernet-Schnittstelle auf der ALU-Unterseite an oder nutzen Sie eine bestehende GPRS-Verbindung.
- Geben Sie im Windows Startmenü im Dialog Ausführen *telnet 192.168.0.32* ein (falls Sie die ALU-IP-Adresse geändert haben, geben Sie die geänderte Adresse ein).

In Dialogfenster das anschließend erscheint wird die WindowsCE Version und die Firmware-Version angezeigt.



Hinweis: Die im Folgenden dargestellten Anzeigen sind nur Beispiele und können je nach ALU-Konfiguration von den tatsächlichen Anzeigen abweichen.

services list

Über diesen Aufruf können Sie feststellen, welche Treiber und Routinen auf der ALU gestartet sind.

Beispiel:

```
HTP0:    0x00030110      HTTPD.DLL                                Running
TEL0:    0x00031e90      TELNETD.Dll                              Running
FTP0:    0x00036190      FTPD.Dll                                 Running
AUC0:    0x00036350      Storage Card\A320_SYS\AX20UDPCFG.dll     Running
S4U0:    0x0003c620      \Storage Card\PCOS_SYS\SVC_SEK_I104SUS.dll Running
AXR1:    0x00036d90      Storage Card\A320_SYS\AXRASVPN.dll       Running
R4U1:    0x0003c7b0      Storage Card\PCOS_SYS\SVC104SA.dll       Running
```

diag_cpu

Zeigt ihnen Betriebsystem-, Hardware- und Treiber-Versionen an.

```
============================================
OS Version          : 4.20 Build 0 Pb-Id 3
ALU Version         : 3.23 Build 0.1  09.11.09
PLC SYSTEM          : A120
PLC ALU             : ALU 320
ProConOS Version    : 4.40
Pcos Version        : 3.16
Primary class name  : CPU
Name of the board   : XBD1
Manufacturer name   : OHP GmbH / JUMP
Date                : 30.1.2009
Serial number       : UAD6K0200
Hardware revsion    : 0.1
Firmware revsion    : 1.5
```

```
Jida interface rev.: 1.2
Jida boot count    : 192
--------------------------------------------
```

diag_rasvpn /svc

Zeigt Ihnen die RASVPN Statistik an.

```
        RASVPN Status: (11) RASVPN service - VPN ready
        RASVPN Nmb RasFct() Error: 1
        Modem Nmb Reset : 7
        PPP -ISP Nmb Dial:    11         (Einwahlversuche)
        PPP -ISP Nmb Connect: 10         (erfolgreiche Verbindungen)
        PPTP-VPN Nmb Login:   41         (VPN-Aufbau-Versuche)
        PPTP-VPN Nmb Connect: 30         (erfolgreiche VPN-Verbindungen)
        RASVPN Nmb RasFct() Error: 1     (letzter aufgetretener Fehlercode)
        Modem Nmb Reset : 7              (Anzahl der Modem-Resets)
```

Es wird jeweils die Anzahl seit dem letzten Hochlauf der ALU erfolgten Verbindungen etc. angezeigt.

diag_rasvpn /stats

Zeigt Ihnen Informationen bezüglich der RAS-Verbindungen an.

```
 RAS - 2 Active connections
 01.) 'RASPPTP'         Status='Connected'
        DevName='RAS VPN Line 0' DevType='vpn'
        Bytes S=10903, R=11056
        Frames S=199, R=258
        Compression I=100% O=107%
        Duration 221:46.766 [mm:ss.mss]
        Link (start):  28800 bps
 02.) 'RASMODEM'        Status='Connected'
        DevName='Hayes Compatible on COM2:' DevType='modem'
        Bytes S=62180, R=59404
        Frames S=1139, R=1144
        Compression I=0% O=0%
        Duration 223:52.944 [mm:ss.mss]
        Link (start):  28800 bps
```



Außerdem stehen Windows-Standardfunktionen wie z.B. ipconfig zur Verfügung.

Beispiel bei einer besehenden GPRS-Verbindung

ipconfig

```
Windows IP configuration

Ethernet adapter [AX887961]:
         IP Address ........ : 0.0.0.0
         Subnet Mask ....... : 0.0.0.0

Ethernet adapter [SERIAL ON COM1:]:
         IP Address ........ : 10.209.197.137
         Subnet Mask ....... : 255.0.0.0
         Default Gateway ... : 10.209.197.137

Ethernet adapter [RAS VPN LINE 0]:
         IP Address ........ : 13.137.0.116
         Subnet Mask ....... : 255.255.255.255
         Default Gateway ... : 13.137.0.116

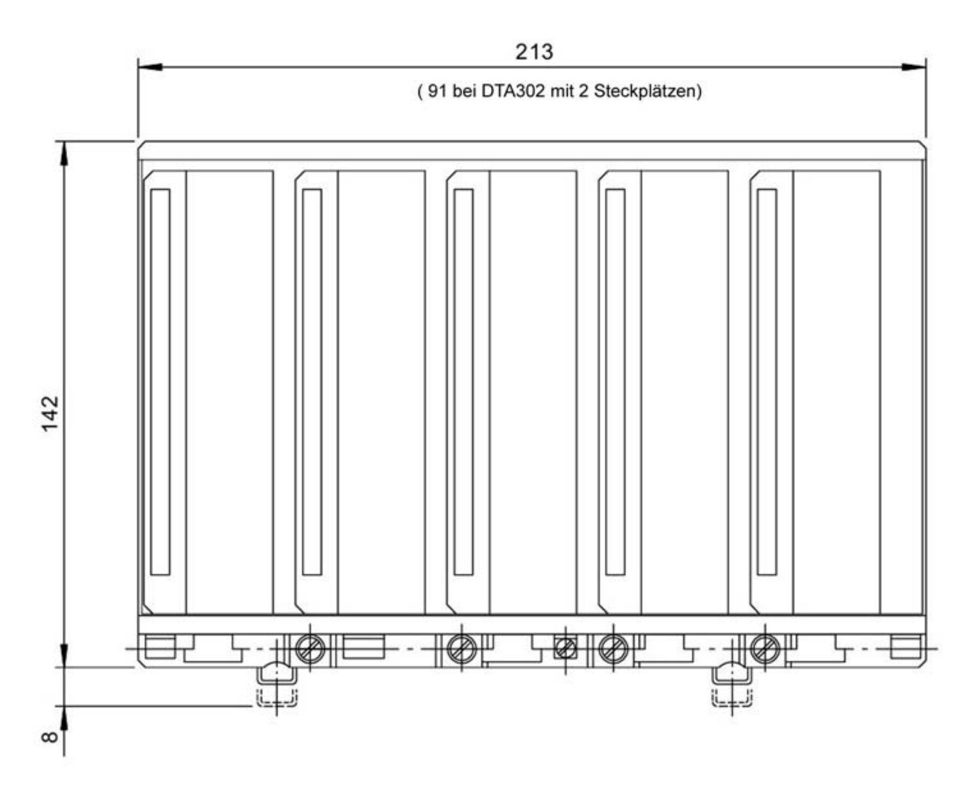
         DNS Servers........ : 10.74.83.22
                               193.254.160.1
```

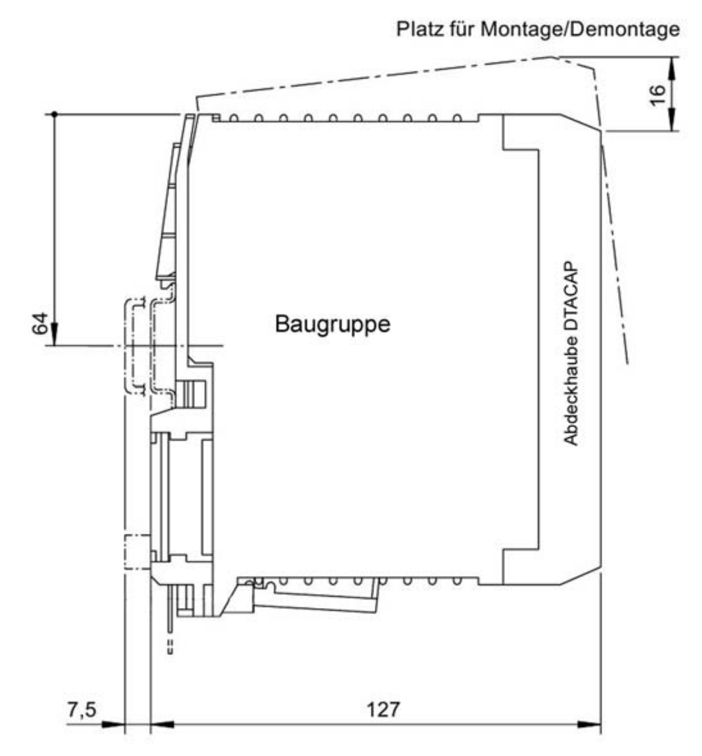


Teil III Baugruppenbeschreibungen

1 Montageabmessungen

Abmessungen DTA300 und DTA301

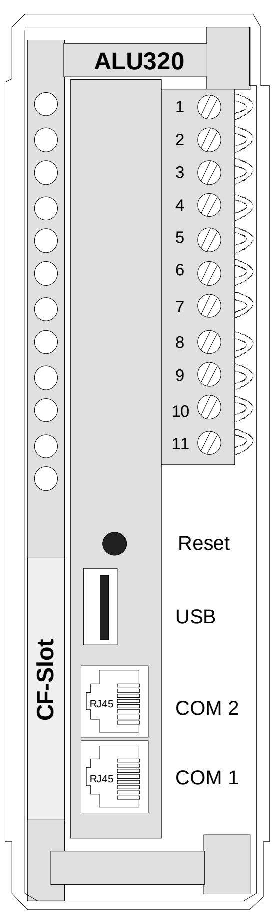






2 Baugruppenbeschreibung ALU 320

2.1 Schnittstellen



ALU 320	
Power	U
Run	U
intern	
TxD2	
RxD2	
TxD1	
RxD1	
Link	
Active	M
	M

Bezeichnung	Steckverbinder	Platzierung
1 Ethernet, Programmierschnittstelle *)	RJ 45	Unterseite
2 Serielle Schnittstellen RS232	RJ 45	Frontseite
Diverse Anschlüsse siehe Kap. 1.2	Schraubklemmen	Frontseite
CF-Steckplatz	3M low Profile	Frontseite
1 USB Schnittstelle	1 USB Stecker	Frontseite
1 VGA Schnittstelle	15pol. Dsub HD-Stecker	Unterseite
1 Antennenanschluss für GPRS (nur ALU320-017)	SMA (f)	Oberseite
1 Steckplatz für SIM-Karte (nur ALU320-017)	**)	Oberseite

*) **Hinweis:** Bei Auslieferung ist die Ethernet-Schnittstelle auf IP-Adresse 192.168.0.32 voreingestellt.

) **Hinweis: Bei ALU320-0017 wird die SIM-Karte so eingesteckt, dass die abgeschrägte Kante nach oben und zur Rückseite der ALU320 zeigt. Der PIN-Code der SIM-Karte muss deaktiviert werden.

2.1.1 Schnittstellenbelegung der Schraubklemmen:

Klemme	Funktion	Beschreibung
1	UB	24V Betriebsspannung
2	UB	24V Betriebsspannung
3	Reserviert	
4	CAN_HZ	CAN High Zul. *)
5	CAN_LZ	CAN Low Zul. *)
6	CAN_GND	CAN Ground *)
7	CAN_GND	CAN Ground *)
8	Reserviert	
9	Reserviert	
10	GND	Ground Betriebsspannung
11	GND	Ground Betriebsspannung

*) in Vorbereitung

2.1.1.1 Stromversorgung

Die ALU320 besitzt ein internes Netzteil mit einem Eingangsspannungsbereich von 14-30 VDC. Hierüber werden die ALU320 und die externen Peripheriekarten versorgt. Für die Peripheriebaugruppen die über den PAB1 angeschlossen werden, stehen 3500 mA zur Verfügung.

Im Unterspannungsfall < 14 V wird der Signalspeicher und der Telegrammpuffer der CPU auf der CF Card nullspannungssicher abgelegt.

Es wird empfohlen die ALU320 mit einem galvanisch getrennten Netzgerät mit mindesten 5 A zu versorgen.

2.1.1.2 Schnittstelle für einen Optionalen GPS Uhrzeitempfänger

Die ALU320 besitzt eine Anschlussmöglichkeit für den optionalen GPS Uhrzeitreceiver GPSRCV von OHP.

Dieser wird über eine serielle Schnittstelle an die ALU320 angeschlossen.

2.1.1.3 CAN Schnittstelle (in Vorbereitung)

Die ALU320 besitzt eine CAN Master/Slave – Schnittstelle. Die Anschaltung basiert auf dem Philips Chip SAJ1000. Die Schnittstelle ist mit Optokopplern galvanisch getrennt. Die Busadresse und die Terminierung werden über den 8-pol. DIL Schalter auf dem linken Seitenteil der ALU 320 eingestellt.

(siehe ALU320 Bedienelemente)



2.1.2 Serielle RS232 Schnittstellen:

Pin-Nr.	Kurz Bez.	Signalbezeichnung	DIN-Bez.	DIN-Signalbezeichnung
1	DCD	Data Carrier Detect	M5	Empfangssignalpegel
2	RXD	Receive Data	D2	Empfangsdaten
3	TXD	Transmit Data	D1	Sendedaten
4	DTR	Data Terminal Ready	S1.2	DEE betriebsbereit
5	GND	Ground	E2	Betriebserde
6	+5VDC	Versorgung für GPS-Empfänger		
7	RTS	Request to Send	S2	Sendeteil anschalten
8	CTS	Clear to Send	M2	Sendebereitschaft

2.1.3 RS485 Schnittstelle bei ALU320-485:

Pin-Nr.	Kurz Bez.	Signalbezeichnung	DIN-Bez.
12	T+	Sendedaten (+)	TXD +
13	T-	Sendedaten (-)	TXD -
14	R-	Empfangsdaten(-)	RXD -
15	R+	Empfangsdaten(+)	RXD +
22		Ground	GND

2.1.4 Compact Flash Speicherkarten

Die ALU 320 verfügt über einen Speicherkartensteckplatz, für eine CF Card. Die Karte wird unterhalb der LEDs eingesteckt. Es können Karten mit max. 8 GB Speichergröße verwendet werden.

Die CF Card enthält alle relevanten Daten für die ALU320, wie z.B. Betriebs- und Laufzeitsystem, Anwenderapplikation und Telegrammpuffer. Die ALU320 wird mit einer betriebsfertigen CF- Card ausgeliefert.

Achtung: Die CF-Karte nur im spannungslosen Zustand ziehen oder stecken. Formatieren Sie die von OHP ausgelieferten CF-Karten nicht, da hierdurch das Betreibsystem Windows CE gelöscht wird. Kopieren bzw. überschreiben Sie auch niemals die Datei CEBOOT.SYS im Stammverzeichnis der CF-Karte.

2.1.5 ALU320 Anzeigeelemente

Die ALU320 enthält 12 LEDs die den Betriebszustand der ALU320 anzeigen.

Nr.	Funktion	Farbe	Anzeige
1	UB OK	grün	Dauerlicht
2	AWP Run, Stopp, Störung	grün	Dauerlicht, Dunkel, Blinkt
3	Zustand interne COM3	grün	Dauerlicht, Dunkel, Blinkt
4	Reserve	rot	
5	TXD2	grün	Blinkt
6	RXD2	gelb	Blinkt
7	TXD1	grün	Blinkt
8	RXD1	gelb	Blinkt
9	Ethernet Link	grün	Blinkt
10	Ethernet Active	gelb	Dauerlicht
11	Reserve	grün	
12	Reserve	rot	

Währen der Hochlaufphase nach Spannungs-EIN durchlaufen die LEDs mehrere Blink/Dauerlichtzyklen.

Nach Ende der Hochlaufphase zeigen die LED folgendes Bild:

- LED 1 Dauerlicht
- Bei angeschlossenem Ethernet LED 9 blinkt, LED 10 Dauerlicht
- Ist ein Anwenderprogramm als Bootprojekt auf der CF Card gespeichert wird dieses gestartet, LED 2 hat dann Dauerlicht. Ist kein Anwenderprogramm geladen, ist die LED 2 aus.

Bedeutung LED 3 wenn eine ALU mit integriertem Modem für den Wählbetrieb verwendet wird.

LED	Bedeutung
Dunkel	Keine Verbindung
Dauerlicht	Datenverbindung, Passwort OK, IEC-Kommunikation
Blinken langsam	Verbindungsaufbau eingeleitet durch ALU
Blinken schnell	Verbindungsaufbau eingeleitet durch Gegenstelle (RING)
2 mal Blinken (500ms) - Pause	Modem Initialisierungsproblem
3 mal Blinken (300ms) - Pause	Modem nicht mehr bereit

Anmerkung: Die Funktion steht erst ab ALU-Firmwareversion 1.91 zur Verfügung.



2.1.6 ALU320 Bedienelemente

Reset Taster

Die CPU Platine enthält einen Reset-Taster auf der Vorderseite. Dieser Taster hat unabhängig von der CPU folgende Funktion:

- drücken des Tasters > 5s: Hardware Reset

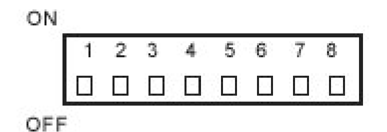
- drücken des Tasters < 5s: Auslösung eines freien Interrupts

DIL Schalter

Der 8 pol. DIL-Schalter dient zur Einstellung des Startverhaltens sowie der CAN-Busadresse und der CAN Bus Terminierung (CAN Bus in Vorbereitung).

Der Schalter ist auf der Lötseite montiert und kann über eine Ausfräsung der linken Gehäuseseite der ALU120 bedient werden.

Schalter	Funktion
1	CAN Adr. (in Vorbereitung)
2	CAN Adr. (in Vorbereitung)
3	CAN Adr. (in Vorbereitung)
4	CAN Adr. (in Vorbereitung)
5	Reserviert
6	**ON = Warmstart, OFF = Kaltstart**
7	ON = CAN Bus abgeschlossen
8	ON = CAN Bus abgeschlossen



2.2 Technische Daten ALU320

Versorgung	
Versorgungsspannung Verlustleistung ALU320 Verfügbarer Strom für E/A Teilnehmer	14V – 30V < 5 W, ohne Erweiterungskarte max. 3500mA
Serielle Schnittstellen	
Anzahl Art, Geschwindigkeit Anschluss	2 seriell, asynchron, V.24 Pegel, max. 19.200 Baud RJ45, Anschluss frontseitig
Ethernet	
Anzahl Art Anschlussstecker	1 IEEE802.3, 100 Base T, 100 MBit/s RJ45, Anschluss von Geräteunterseite
USB	
Anzahl Art	1 x USB 1.1 1 Standard USB Frontseitig
Steckplatz für CF Speicherkarte	
Anzahl Art	1 Standard Compact Flash (CF Card), max. 8 Gbyte
CAN Anschluss	
Anzahl, Art Anschluss	1 CAN Master, Anschluss über Schraubklemmen frontseitig.
CPU /Speicher	
Prozessortyp Speicher Telegrammpuffer Betriebssystem	skalierbares CPU Board 32 MByte RAM / 32 MByte Flash 1 MB im Flash, keine Batterie erforderlich Windows CE 4.2 Echtzeitkernel
Uhrzeitbehandlung	
Art Synchronisation	Gepufferte ms genaue RTC Über optionales GPS Modul über serielle Schnittstelle
CPU Erweiterungssteckplatz	
Anzahl, Art	1 Steckplatz für Modem Erweiterungskarten
Schutzart	
Schutzart (IEC 60529) Schutzklasse (IEC61140) Isolation	IP20 III EN60950, IEC 950
Umweltbedingungen	
Umgebungstemperatur bei Betrieb Lagertemperatur Rel. Feuchte (IEC 68-2-1-1/2)	0°C bis 60°C -25°C bis 85°C bis 95% keine Betauung
Anzeigen	
9 LED-Anzeigen	1 LED UB OK 2 LED run, Stopp, Störung 3 LED interne COM-Schnittstelle 4 LED res. 5 LED TXD1 6 LED RXD1 7 LED TXD2 8 LED RXD2 9 LED Ethernet Link 10 LED Ethernet Active
Mechanischer Aufbau	
Format	Standard A120/Compact E/A Teilnehmer 3 HE, 8TE